

**NUREG/CR-6525**  
**SAND93-4032**

---

---

# **SECPOP90: Sector Population, Land Fraction, and Economic Estimation Program**

---

---

Prepared by  
S L Humphreys  
J A Rollstin  
J. N. Ridgely

**Sandia National Laboratories**

**Operated by**  
Lockheed Martin Corporation

**Gram, Inc.**

**Prepared for**  
**U.S. Nuclear Regulatory Commission**



## AVAILABILITY NOTICE

### Availability of Reference Materials Cited in NRC Publications

Most documents cited in NRC publications will be available from one of the following sources:

1. The NRC Public Document Room, 2120 L Street, NW , Lower Level, Washington, DC 20555-0001
2. The Superintendent of Documents, U S Government Printing Office, P O Box 37082, Washington, DC 20402-9328
3. The National Technical Information Service, Springfield, VA 22161-0002

Although the listing that follows represents the majority of documents cited in NRC publications, it is not intended to be exhaustive.

Referenced documents available for inspection and copying for a fee from the NRC Public Document Room include NRC correspondence and internal NRC memoranda; NRC bulletins, circulars, information notices, inspection and investigation notices; licensee event reports; vendor reports and correspondence; Commission papers; and applicant and licensee documents and correspondence

The following documents in the NUREG series are available for purchase from the Government Printing Office: formal NRC staff and contractor reports, NRC-sponsored conference proceedings, international agreement reports, grantee reports, and NRC booklets and brochures Also available are regulatory guides, NRC regulations in the *Code of Federal Regulations*, and *Nuclear Regulatory Commission Issuances*

Documents available from the National Technical Information Service include NUREG-series reports and technical reports prepared by other Federal agencies and reports prepared by the Atomic Energy Commission, forerunner agency to the Nuclear Regulatory Commission

Documents available from public and special technical libraries include all open literature items, such as books, journal articles, and transactions *Federal Register* notices, Federal and State legislation, and congressional reports can usually be obtained from these libraries

Documents such as theses, dissertations, foreign reports and translations, and non-NRC conference proceedings are available for purchase from the organization sponsoring the publication cited

Single copies of NRC draft reports are available free, to the extent of supply, upon written request to the Office of Administration, Distribution and Mail Services Section, U S Nuclear Regulatory Commission, Washington, DC 20555-0001.

Copies of industry codes and standards used in a substantive manner in the NRC regulatory process are maintained at the NRC Library, Two White Flint North, 11545 Rockville Pike, Rockville, MD 20852-2738, for use by the public. Codes and standards are usually copyrighted and may be purchased from the originating organization or, if they are American National Standards, from the American National Standards Institute, 1430 Broadway, New York, NY 10018-3308

## DISCLAIMER NOTICE

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for any third party's use, or the results of such use, of any information, apparatus, product, or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights.

**NUREG/CR-6525  
SAND93-4032**

---

---

# **SECPOP90: Sector Population, Land Fraction, and Economic Estimation Program**

---

---

Manuscript Completed July 1997  
Date Published: September 1997

Prepared by  
S. L. Humphreys<sup>a</sup>  
J. A. Rollstin<sup>b</sup>  
J. N. Ridgely<sup>c</sup>

<sup>a</sup> Sandia National Laboratories  
Albuquerque, NM 87185-0748

Subcontractor.

<sup>b</sup> GRAM, Inc  
8500 Menaul Blvd., NE  
Albuquerque, NM 87112

<sup>c</sup> J. N. Ridgely, NRC Project Manager

**Prepared for**  
**Division of Systems Technology**  
**Office of Nuclear Regulatory Research**  
**U.S. Nuclear Regulatory Commission**  
**Washington, DC 20555-0001**  
**NRC Job Code J6024**



Microsoft and MS-DOS are registered trademarks of the Microsoft Corporation  
Microsoft Windows 3.1, Windows 3.11, Windows NT, Windows 95, Microsoft Visual Basic, Microsoft QuickBasic,  
Microsoft Excel, and Microsoft Access are trademarks of the Microsoft corporation.  
IBM, PS/2, Personal Computer AT, and OS/2 are trademarks of the IBM Corporation  
Intel and 80386 are trademarks of the Intel Corporation  
MapPlan is a trademark of WordTech Systems, Incorporated.  
Pizazz Plus is a registered trademark of Application Techniques, Incorporated  
HiJaak PRO is a registered trademark of the Quarterdeck Corporation.  
dBase and dBase III is a registered trademark of Borland International, Incorporated

## ABSTRACT

In 1973 Mr. W. Athey of the Environmental Protection Agency wrote a computer program called SECPOP which calculated population estimates. Since that time, two things have changed which suggested the need for updating the original program — more recent population censuses and the widespread use of personal computers (PCs). The revised computer program uses the 1990 and 1992 Population Census information and runs on current PCs as "SECPOP90".

SECPOP90 consists of two parts, site and regional. The site analysis provides population and economic data estimates for any location within the continental United States. Siting analysis is relatively fast running. The regional portion assesses site availability for different siting policy decisions, i.e., the impact of available sites given specific population density criteria within the continental United States. Regional analysis is slow.

This report compares the SECPOP90 population estimates and the nuclear power reactor licensee-provided information. Although the source, and therefore the accuracy, of the licensee information is unknown, this comparison suggests SECPOP90 makes reasonable estimates. Given the total uncertainty in any current calculation of severe accidents, including the potential offsite consequences, the uncertainty within SECPOP90 population estimates is expected to be insignificant.



## CONTENTS

<b>ABSTRACT</b> . . . . .	iii
<b>ACKNOWLEDGMENTS</b> . . . . .	xi
<b>1 INTRODUCTION</b> . . . . .	1
1.1 Report Conventions . . . . .	1
1.2 Background. . . . .	1
1.3 Hardware and Software Requirements . . . . .	2
<b>2 INSTALLATION</b> . . . . .	3
2.1 Initialization . . . . .	3
2.2 Starting SECPOP90 . . . . .	3
<b>3 USER'S GUIDE</b> . . . . .	5
3.1 Parts of the Main Screen . . . . .	5
3.2 Main Menu . . . . .	6
3.3 Site Definition . . . . .	7
3.3.1 New Site . . . . .	7
3.3.2 Open Site . . . . .	9
3.3.3 Save Site . . . . .	10
3.3.4 Save As Site . . . . .	10
3.4 Problem Definition . . . . .	10
3.4.1 New Problem. . . . .	11
3.4.2 Economic Regions . . . . .	13
3.4.3 Open Problem . . . . .	15
3.4.4 Save Problem . . . . .	16
3.4.5 Save As Problem . . . . .	16
3.5 Performing Site-Specific Calculations . . . . .	17
3.6 Site-Specific Results . . . . .	19
3.6.1 Population Rosette. . . . .	19
3.6.2 Population Tables . . . . .	19
3.6.3 MACCS Input File . . . . .	21
3.6.4 Save Results . . . . .	24
3.6.5 Load Previous Results . . . . .	24
3.7 Performing Regional Calculations . . . . .	25
3.8 Regional Calculation Results . . . . .	27
3.9 Customizing SECPOP90 . . . . .	29
3.9.1 Setup Form Fields . . . . .	30
3.9.2 Setup Form Commands . . . . .	30
3.10 Exiting SECPOP90 . . . . .	31

3 10.1 DOS Shell	32
3 10.2 Exit	32
<b>4 COMPUTATIONAL METHODOLOGY</b>	<b>33</b>
4.1 Population Count and Density Algorithms	33
4.1.1 Boundaries Algorithm	33
4.1.2 First Element Location Algorithm	34
4.1.3 Specific Grid Element Determination	34
4.2 Land Fraction Algorithms	35
4.3 Economic Factors Algorithms	35
4.4 Database Structure	36
4.4.1 Block-Level Database	36
4.4.2 County-Level Database	36
<b>5 SOFTWARE DEVELOPMENT</b>	<b>39</b>
5.1 Block-Level Census Database	39
5.1.1 Construction of the Block-Level Census Database	39
5.1.2 Verification and Validation of the Block-Level Census Database	42
5.2 County-Level Census Database	42
5.2.1 Construction of the County-Level Census Database	42
5.2.2 Verification and Validation of the County-Level Census Database	44
5.3 Verification and Validation of the SECPOP90 Code	44
5.3.1 Internal Verification and Validation	44
5.3.2 External Verification and Validation	45
<b>APPENDIX A — SITE LIST</b>	<b>A-1</b>
<b>APPENDIX B — MACCS SITE DATA FILE FORMAT</b>	<b>B-1</b>
<b>APPENDIX C — COUNTY DATABASE LISTING</b>	<b>C-1</b>
<b>APPENDIX D — STATISTICAL ABSTRACTS</b>	<b>D-1</b>
<b>APPENDIX E — OTHER CENSUS INFORMATION</b>	<b>E-1</b>
<b>APPENDIX F — POPULATION TABLES</b>	<b>F-1</b>
<b>APPENDIX G — CONVERSION PROGRAM LISTINGS</b>	<b>G-1</b>
<b>APPENDIX H — TEST PROGRAM LISTINGS</b>	<b>H-1</b>
<b>APPENDIX I — SECPOP90 PROGRAM LISTINGS</b>	<b>I-1</b>

## FIGURES

Figure 2 1 File and Directory Structure .....	3
Figure 2 2 Initial Screen .....	4
Figure 3.1 Parts of the Main Screen .....	5
Figure 3.2 Main Menu .....	6
Figure 3 3 Site Sub-Menu .....	7
Figure 3 4 Site Form .....	7
Figure 3 5 Outside of the Continental U S .....	8
Figure 3 6 Print Form .....	9
Figure 3 7 Open Site Form .....	9
Figure 3 8 Previous Site Data .....	10
Figure 3.9 Save As Site Form .....	10
Figure 3 10 Problem Sub-Menu .....	11
Figure 3 11 Problem Form .....	11
Figure 3.12 Rosette Definition .....	12
Figure 3 13 Radial Unit Conversion .....	12
Figure 3.14 Economic Regions Form .....	13
Figure 3.15 Example Economic Regions Form .....	14
Figure 3 16 Example Economic Regions Rosette .....	14
Figure 3.17 Open Problem Form .....	16
Figure 3 18 Previous Problem Data .....	16
Figure 3.19 Save As Problem Form .....	17
Figure 3.20 Calculate Sub-Menu .....	17
Figure 3 21 Calculation Error .....	17
Figure 3 22 Initial Calculate Form .....	17
Figure 3 23 Subsequent Calculate Forms .....	18
Figure 3 24 Stop Calculation .....	19
Figure 3.25 Results Sub-Menu .....	19
Figure 3 26 Population Rosette .....	20
Figure 3 27 Population Tables Sub-Sub-Menu .....	20
Figure 3.28 Population Table Form .....	21
Figure 3 29 Cumulative Population Table Form .....	21
Figure 3 30 MACCS Input File Sub-Sub-Menu .....	22
Figure 3 31 MACCS Site Data File Form .....	22
Figure 3 32 Save Data Form .....	24
Figure 3.33 Load Previous Results Form .....	24
Figure 3 34 Calculate Sub-Menu .....	25

Figure 3 35 Regional Calculation Setup Form.....	25
Figure 3 36 Radii and Longitudinal Spacing .....	26
Figure 3.37 Regional Calculation Percent Complete .....	27
Figure 3 38 Regional Calculation Total Time .....	27
Figure 3.39 Results Sub-Menu.....	27
Figure 3.40 Output Map .....	28
Figure 3 41 MapPlan File Sub-Menu .....	28
Figure 3 42 Export CGM File Form.....	29
Figure 3.43 Print Current Image Form.....	29
Figure 3 44 Print Setup Form .....	29
Figure 3 45 Setup Form .....	30
Figure 3 46 MapPlan Setup Form .....	31
Figure 3 47 Exit Sub-Menu .....	31
Figure 3 48 The DOS Shell.....	32
Figure 3 49 Exit Confirmation Form .....	32
Figure 5 1 PL94-71 README File dBase to ASCII Conversion Program .....	41
Figure 5 2 Millstone Rosette .....	46
Figure 5 3 Monticello Rosette...	46
Figure 5 4 Diablo Canyon Rosette .....	47

## TABLES

Table 1.1 Report Conventions. ....	1
Table 3 1 Example MACCS Site Data File	22
Table 3.2 MACCS and CSV Formats ..	31
Table 4.1 Excerpt From the County-level Database	37
Table 5.1 dBase III Census Record Format	39
Table 5 2 Census of Agriculture Data Items ..	43
Table 5 3 Total Reported and Calculated Populations ..	45



## **ACKNOWLEDGMENTS**

The SECPOP90 team thanks everyone who helped make this report and the SECPOP90 software a reality. In particular we are especially grateful to Brian Lewis, our summer student from Cornell University, who helped make SECPOP90 a more useful program by providing it with many additional features and extensive error checking of all user input, and Donald Mitchell, Sandia National Laboratories, whose assistance with the county-level database effort was invaluable.



# 1 INTRODUCTION

This report is divided into five sections. The first section is this introduction. The second section describes the installation of the SECPOP90 software. The third section is the user's guide, which explains how to use the software in a step-by-step manner. The fourth section describes the algorithms and methodologies used to perform the population, land fraction, and economic estimates. The fifth section discusses the software design, development, and verification and validation. Additional appendices describe input and output file formats, the U.S. Bureau of Census CD-ROMs to block census and county census data file conversions, and the complete SECPOP90 Visual Basic code listing.

## 1.1 Report Conventions

The following conventions will be used throughout this report.

**Table 1.1 Report Conventions**

Convention	Explanation
SECPOP90 V2.0 MACCS Site Data File	A monospaced font will be used for example data files, operating system messages, code fragments, and the SECPOP90 code listings
<b>a\install</b>	A bold monospaced font will be used for input that the user is expected to type in
←	The return or enter key. The user should press this key where it is indicated in this report
<i>Filename</i>	An italicized monospaced font will be used for input parameters that the user is to provide and type in
HOME, END, PAGE UP	Text in small caps indicates one of the special keys on the keyboard.
ALT+F1	A plus sign indicates that the first key should be pressed and held down then the second and subsequent keys pressed
Three-Letter Acronym (TLA)	Acronyms will be defined where they are first used. The defining phrase will be capitalized in correspondence to the letters that make up the acronym. The definition will be followed by the acronym in parentheses

## 1.2 Background

In 1973 Mr. W. Athey of the Environmental Protection Agency wrote a computer program called SECPOP which calculated population estimates. This program was for use on mainframe computers, used the 1970 population census data, and was originally part of a study on air quality. The Nuclear Regulatory Commission used this program as part of the siting review for nuclear power plant construction and operating licensing applications.

Since then, two things have changed which suggested the need for updating the original program — more recent population censuses and the widespread use of personal computers (PCs). The revised computer program uses the 1990 Population Census information and runs on current PCs as "SECPOP90".

SECPOP90 consists of two parts site and regional. The site analysis provides population and economic data estimates for any location within the continental United States. Calculation results can be displayed, printed, or stored as a file as a rosette, tables, or a MELCOR Accident Consequence Code System (MACCS) site file. Site analysis is relatively fast running. The regional portion assesses site availability for different siting policy decisions, i.e., to assess the impact of available sites given specific population density criteria within the continental United States. Regional analysis is much slower.

SECPOP90 uses the latest (1990) census data and provides much greater resolution than the 1980 census data. This resolution is achieved by utilizing over six million census data points and performing the sector population estimates using these points directly. SECPOP90 also allows the user to calculate regional population density thresholds and display them via an interface with MapPlan mapping software.

### 1.3 Hardware and Software Requirements

SECPOP90 will run with most personal computers available today. SECPOP90 operates as a stand-alone program in the DOS environment. It also runs in a DOS box in OS/2 Version 2 or Windows Version 3.1 or later. Due to the operational overhead associated with Windows, SECPOP90 will run slower under Windows.

SECPOP90 consists of 422 programs and files requiring approximately 81 megabytes of hard disk storage. This is adequate for running only the "site" portion of SECPOP90. The "regional" calculation can require significantly more hard disk storage space. An additional 35 megabytes may be needed to analyze the entire continental United States in 10 mile rings. One calculation (US NRC Region I with 10 mile increments) took approximately 14 days and left a file of almost 6 megabytes on the hard disk. (More about "site" and "regional" calculations in sections 3.2 and 3.3, respectively.)

An American National Standards Code for Information Interchange (ASCII) compatible printer is required in order for the SECPOP90 print functions to print properly. In addition to printing the standard ASCII characters, it must also be capable of executing a form feed when receiving the ASCII form feed character (CTRL+L).

#### Minimum System Requirements

IBM PS/2, IBM Personal Computer AT or compatible,

IBM PC-DOS 5.0, Microsoft MS-DOS 5.0 or compatible,

Intel 80386, or compatible CPU,

620 kilobytes of available conventional RAM, 512 kilobytes expanded RAM,

One 3½ inch high density (1.44 megabyte) floppy disk drive,

Hard disk with 81 megabytes free (minimum to load programs, 100 megabytes recommend),

VGA monitor, or compatible, and

A Microsoft-compatible mouse (recommended but not required)

## 2 INSTALLATION

This section describes how to install SECPOP90 under DOS. To install SECPOP90 under Windows 3.1x, Windows 95, Windows NT, or OS/2, install SECPOP90 in a DOS-compatible window using the instructions provided below. For other operating systems, consult your operating system manual on how to install DOS programs.

To install SECPOP90, first exit or quit all other programs and command shells until you reach the DOS prompt (for example C:\>) Then insert disk # 1 into drive A or B and type

**a:\install ↵**

or

**b:\install ↵**

respectively

Follow the on-screen prompts to insert the additional disks. When the installation is complete the directory will contain the following files and sub-directories shown in Figure 2.1. (The default directory is \SECPOP90. The root of the directory will be different if you chose another directory during installation.) For more information about these files see sections 3.7, 4.4, and Appendix A.

### 2.1 Initialization

All SECPOP90 output files can be printed directly from SECPOP90 or from any word processor or other program capable of printing ASCII text files. The forms themselves and the population rosette (see section 3.6.1) cannot be directly printed. To print or save the forms, the population rosette, or MapPlan output files, third-party software capable of performing a graphics screen capture (for example, Pizazz Plus or HiJaak PRO) may need to be loaded before running SECPOP90. Consult the documentation for your screen capture software. If SECPOP90 is being run in the Microsoft Windows environment, then it is possible to print SECPOP90 screen images using ALT+PRINT SCREEN and pasting the image into Microsoft Paintbrush or any Windows program that can print bit mapped (\*.BMP) files. Consult your Windows documentation for more information. Other operating systems might have similar features. Consult your operating system documentation to determine if screen capture is supported.

### 2.2 Starting SECPOP90

To start SECPOP90 at the DOS prompt type

**cd \secpop90 ↵**

to change to the default SECPOP90 directory. If you installed SECPOP90 on a different hard drive or in a different directory, then type

**drive letter: ↵**

**cd \pathname ↵**

where *drive letter* is the hard drive designator (for example, C or D) of the hard drive on which SECPOP90 was installed and *pathname* is the complete pathname of the directory in which SECPOP90 was installed. After successfully changing to the appropriate directory, type the following at the DOS prompt

**secpop90 ↵**

SECPOP90 will now run and display the first screen as shown in Figure 2.2. To start SECPOP90 from other operating systems, consult your operating system manual on how to run DOS programs.

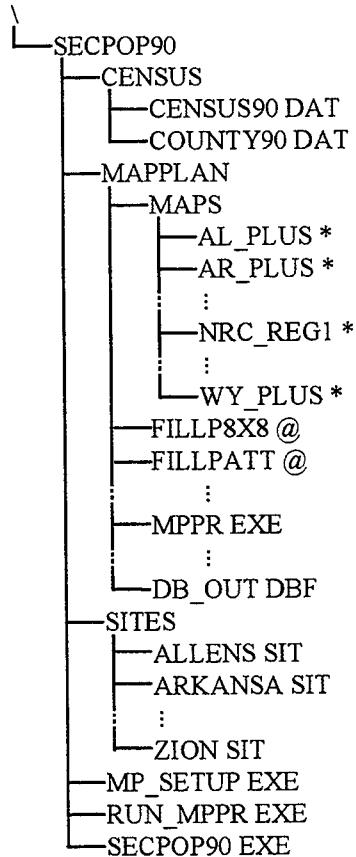


Figure 2.1 File and Directory Structure

SECPOP90 - SECTOR POPULATION and Economic Estimator - Version 2.3  
Site Problem Calculate Results Setup Exit

Welcome to SECPOP90

SECPOP90 calculates estimated population and economic data about any location in the continental United States. It can also estimate regional population densities within the continental United States. All estimates are made using 1990 and 1992 U.S. Bureau of Census data. Population estimates are made using Block level census data. Economic estimates are made using County level census data.

Numerous results from this program have been successfully validated against existing data. However, in no event shall the authors or sponsors of this program be liable for any damages whatsoever arising from the use of, or inability to use, this program, including any conclusions drawn from the results.

Figure 2.2 Initial Screen

### 3 USER'S GUIDE

SECPOP90 consists of two main calculation tasks: calculations related to a specific site and calculations related to a region of the continental United States. The site calculation requires, among other things, a specific latitude and longitude from which to work. This is typically the reactor center for a single unit site or the center point between the reactors at a multi-unit site. (The variance between these points is not expected to significantly affect the results, especially the results of subsequent MACCS calculations.) The regional calculation requires, among other things, the definition of a region of the continental United States. This region can be the entire continental United States or a subset, such as an NRC region, a state, or the area of Washington, DC.

Calculations for both sections start at the initial screen shown in section 3.1. However, SECPOP90 uses a different approach for each type of calculation and each approach is discussed separately. Site-specific calculations are discussed in sections 3.3 through 3.6 and regional calculations are discussed in section 3.7.

Descriptions on how to navigate through the SECPOP90 menus and forms are given in sections 3.2 and 3.3 respectively. If, while trying to perform one of the operations described later in the User's Guide section, you have trouble with keyboard or mouse usage, refer back to these two sections.

#### 3.1 Parts of the Main Screen

The SECPOP90 main screen is divided into three major sections. Section (1), *Main Screen Title Bar*, displays the name of the SECPOP90 program and the present version number. Section (2), *Main Menu*, is the SECPOP90 main menu from which all other menu commands, sub-menus and sub-menu commands are selected and executed. It is through this menu that you access all other portions of the SECPOP90 program. Section (3), *Area for Additional Forms or Messages*, is the area where various other forms and messages will appear. Most of these will require you to provide some information, make a decision, or acknowledge that you have read them. The *Welcome to SECPOP90* form below is the only form that will close automatically when you select a command from the main menu. All other forms will require some input before they will close. Note the *Welcome to SECPOP90* form will also close if you click on it with the mouse or select *Close* from the control box (for more information about the control box, see the *Longitude* field description in section 3.3.1).

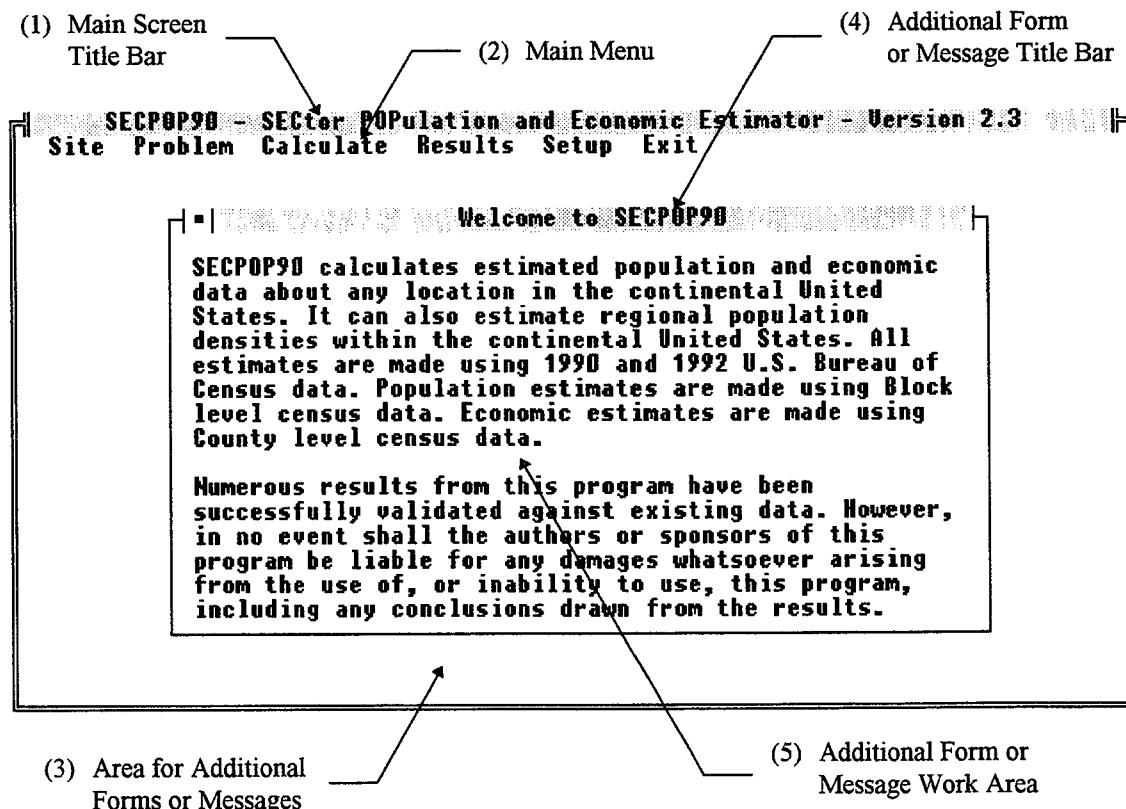


Figure 3.1 Parts of the Main Screen

In addition to the three major sections of the main screen, each additional form has two sections. Section (4), *Additional Form or Message Title Bar*, displays the name of the additional form or message. Each form has a unique name to assist the user so that you know exactly where you are in the program. Section (5), *Additional Form or Message Work Area*, is the area of the additional form or message where fields are filled in, options selected, flags set, and additional commands specific to the form or message may be executed. It is through these additional forms and messages where most of your interaction and work with SECPOP90 will occur.

## 3.2 Main Menu

The main menu consists of six sub-menus or commands

Site	site characteristics for site-specific calculations,
Problem	additional non-site-specific parameters for site-specific calculations,
Calculate	perform site-specific or regional calculations,
Results	results from present or previous calculations,
Setup	customize the SECPOP90 environment, and
Exit	exit SECPOP90

The order of the selections above outlines the order of how you might solve a site-specific problem. First, you define the site parameters such as latitude and longitude using the *Site* sub-menu, next, you define additional problem parameters such as the radius of the exclusion area using the *Problem* sub-menu; then you perform the calculations using the *Calculate* sub-menu; and finally you examine the results using the *Results* sub-menu.

Regional problems do not require any site or problem specifications and are solved by setting the problem parameters such as population density threshold when the calculation is performed using the *Calculate* sub-menu. The results are then shown immediately. Previous regional results may be examined using the *Results* sub-menu.

**Site Problem Calculate Results Setup Exit**

Figure 3.2 Main Menu

The *Setup* command allows you to modify the SECPOP90 environment. Pathnames for site files and flags to control the output may be set in the *Setup* form. The *Exit* sub-menu allows you to execute a DOS shell or exit the SECPOP90 program.

To navigate around the main and sub-menus, there are several methods that can be used. The mouse can be used to reach any command or sub-menu by one of two ways. To use the first method, click (normally with the left button unless you have reversed buttons via your mouse setup software) on the menu item you wish to select. It will then become highlighted (see the selection *Site* in Figure 3.2) and then either the sub-menu will drop down (see the *Site* sub-menu in Figure 3.3) or the command selected will execute. To select a command from a sub-menu, click on the command using the mouse and the command will execute. For sub-sub-menus and sub-sub-menu commands, proceed in a similar fashion. Sub-menu commands and sub-sub-menu commands can also be executed using the mouse by moving the mouse to the menu item for the sub-menu you wish to select (for example, the *Site* menu item if you want the *Site* sub-menu) then press the left (or right, see comment above) mouse button and keep it held down. With the mouse button held down you can trace through the menus, sub-menus, and sub-sub-menus until you get to the command you wish to execute. Release the mouse button when the appropriate command is highlighted and that command will execute.

The menus can also be accessed using the keyboard. Pressing ALT will highlight the main menu bar *Site* selection. The left and right arrow keys can then be used to move the highlight from menu item to menu item. Press ENTER (or RETURN) or the down arrow to select a command or sub-menu. Use the up and down arrows to select commands from the sub-menus or sub-sub-menus. Press ENTER (or RETURN) to execute the desired command. Another way to access the menus is to use the designated "shortcut" keys. Every menu, sub-menu, sub-sub-menu item has a special key associated with it that is highlighted in white (in this report such keys are distinguished in dark gray in the figures and bold at the beginning of each section describing the menu or sub-menu). To use these keys, first press ALT, then press the sequence of shortcut keys that will take you to the command you wish to execute. Typing the last key of the sequence will then execute the command. For example, to begin the site-specifications for a new site type

ALT S N

and the form to define the site parameters for a new site will be displayed. Compare this sequence of letters to the highlighted letters in Figure 3.3 for a clearer understanding of what sequences to use.

### 3.3 Site Definition

Site definition consists of specifying the site name, latitude, longitude, and any desired site-specific remarks. The *Site* sub-menu has four available commands. Note that some of these commands are not available at all times. For example, *Save Site* and *Save As Site* are not enabled if there is no site information to save. The four *Site* sub-menu commands are

- New Site** enter the site-specific parameters for a new site,
- Open Site** open a previously saved site,
- Save Site** save the site information with the present filename; and
- Save As Site** save the site information with a new filename

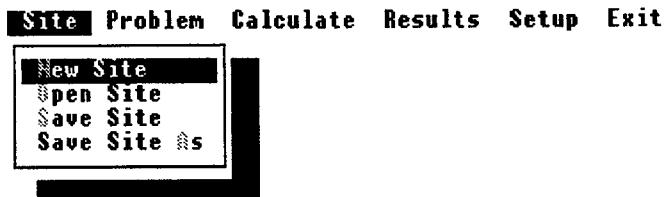


Figure 3.3 Site Sub-Menu

#### 3.3.1 New Site

Selecting *New Site* opens the form shown in Figure 3.4. There are similar methods for moving around the fields (those areas where user input is requested) of the forms as moving through the menus and sub-menus. To move to a field directly with the mouse, move the mouse to the field and click the mouse button. If the field is an option (see the *Radial Units* field description in section 3.4.1), then the operation is complete. If the field is a text field, you may now enter the information requested. If the field is a command button (for example, the *Close* button in Figure 3.4), then the command is executed.

You can also move from field to field using the keyboard. Press ALT, then the "shortcut" key of the field you wish to select, move to, or execute (in this report such keys are distinguished in dark gray in the figures and bold at the beginning of each subsequent section describing the form). You can also move from field to field by pressing TAB or SHIFT+TAB. Note that the arrow keys do not work with forms as they do with menus. For example, to move directly to the *Site Remarks* field of the *Site* form (see Figure 3.4), type

ALT+r

while the *Site* form is showing. Note also that the menus are inactive while forms are showing. To access a menu command all forms must first be closed. Forms and messages can always be closed by pressing ESC.

A screenshot of a Windows-style dialog box for a 'Site' form. The title bar says 'C:\SECP0P90\XITES\GRAND.SIT'. The form contains the following fields:

- Site Name:** A text field containing 'Grand Gulf'.
- Site Coordinates:** A section with two sets of input fields for latitude and longitude in degrees, minutes, and seconds.

	Degrees	Minutes	Seconds		Degrees	Minutes	Seconds
Latitude	32	0	27	Longitude	91	2	53
- Site Remarks:** A large text area for comments.
- Buttons:** A row of buttons at the bottom: 'Sites' (disabled), 'Close' (highlighted in dark gray), 'Save' (disabled), 'Save As' (disabled), and 'Print'.

Figure 3.4 Site Form

## Site Form Fields

The *Site* form (see Figure 3 4) consists of the following fields which are designed so that the user may easily edit all of the site-specific information (fields are designated required or optional depending on whether they are required to perform site-specific calculations):

Title Bar	contains the filename of the site being edited. If this is a new site, then the default filename "NEW_SITE SIT" will be used. If the <i>Site</i> form is edited, then the phrase "(Modified)" will be appended to the site filename. The filename is automatically updated when the user performs a <i>Save</i> or <i>Save As</i> command (see the next section)
Site Name	a descriptive name of the site. Up to eighty characters (both rows of the field in the form) may be used to describe the site. This field will be displayed or printed whenever the site-specific information is displayed or printed. Only the first forty of those characters (the first row) will be printed on the first line of the MACCS Site Data File (see Table 3 1). This field is optional
Latitude	the latitude of the site in degrees, minutes, and seconds. 0s (zeroes) must be entered if the minutes or seconds are equal to 0. These fields are required.
Longitude	the longitude of the site in degrees, minutes, and seconds. 0s (zeroes) must be entered if the minutes or seconds are equal to 0. These fields are required.
	SECPOP90 checks to see if the coordinates of the site lie outside of the continental U.S. by comparing the coordinates to the minimum and maximum coordinates of the continental U.S. If the coordinates are determined to be outside of the continental U.S., then SECPOP90 displays the following error message

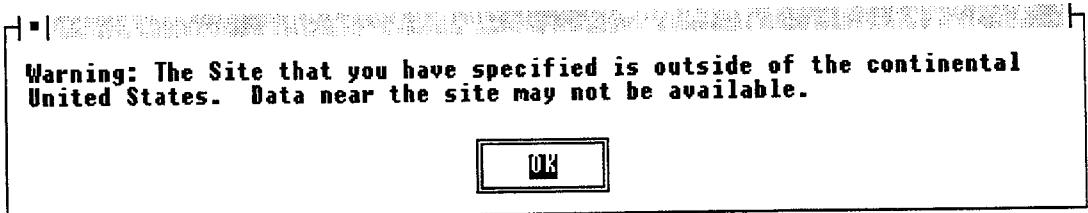


Figure 3.5 Outside of the Continental U.S.

To remove this message select the *OK* command button using one of the above-mentioned techniques for navigating around forms or click the control box (the small square in the upper left-hand corner of the message) and select *Close* from the sub-menu that appears. Other messages can be closed in a similar fashion.

Note this limited checking is meant to alert the user to gross typing errors. The absence of the above message does not guarantee that the site information is correct. The user should always double-check the information in the required fields with the original sources.

Site Remarks	descriptive remarks about the site. Up to eighty characters (both rows of the field in the form) may be used to describe the site. This field will be displayed or printed when viewing this form or printing the site-specific information (see the <i>Print</i> command in the next section). One use of this space could be to record plant specific information. For example
	Unit 1 -- GE BWR 6 with Mark III containment, licensed in 1984

This field is optional

## Site Form Commands

At the bottom of the *Site* form is a row of five command buttons. The commands are executed when the buttons are selected (see 3 3.1). The commands support the opening, closing, saving, and printing of the *Site* form. The command buttons are as follows:

Sites	opens a previously defined site file. This command is equivalent to the <i>Open Site</i> command of the <i>Site</i> sub-menu (see Figure 3 3). For a more detailed explanation see section 3 3.2
Close	closes the <i>Site</i> form. All information on the form is saved in memory and can be reviewed by opening the <i>Site</i> form again. The information will be lost unless the site information is stored in a file (see the <i>Save</i> and <i>Save As</i> commands below) before SECPOP90 is exited

Save saves the site information using the filename displayed in the *Site* form *Title Bar*. This command is equivalent to the *Save Site* command of the *Site* sub-menu

Save As saves the site information using a filename specified by the user. This command is equivalent to the *Save As Site* command of the *Site* sub-menu. For a more detailed explanation see section 3.3.4

Print prints the site-specific information to a connected printer or a file. When this button is selected the following form will be displayed:

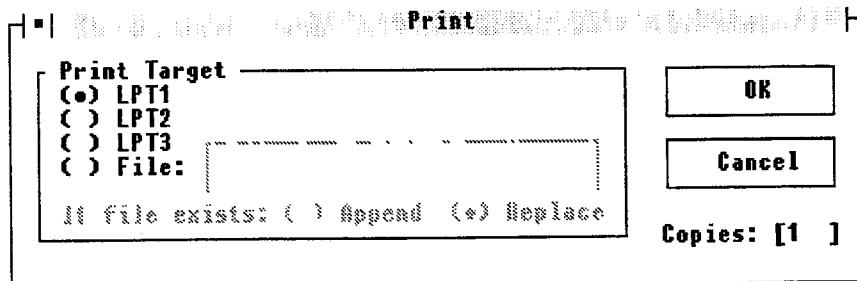


Figure 3.6 Print Form

This form allows the user to select which printer or file the site information is to be printed to. In the case of printing to a file, the user can also select whether to replace an existing file or append this information to the end of it. The user can also specify how many copies of the site-specific information should be printed to either the selected printer or file.

### 3.3.2 Open Site

Selecting *Open Site* opens the form shown in Figure 3.7. This form allows the user to select a previously defined site from either the list of predefined sites (as shown) or his or her own list of sites. The form has fields for selecting the appropriate disk drive, directory, sub-directories, and filename. Once these are chosen and the *OK* button is selected, the file is loaded into memory and the *Site* form will be displayed with the saved information (see Figure 3.4). The user can close the *Open Site* form by selecting *Cancel*, pressing *ESC*, or selecting *Close* from the control box sub-menu.

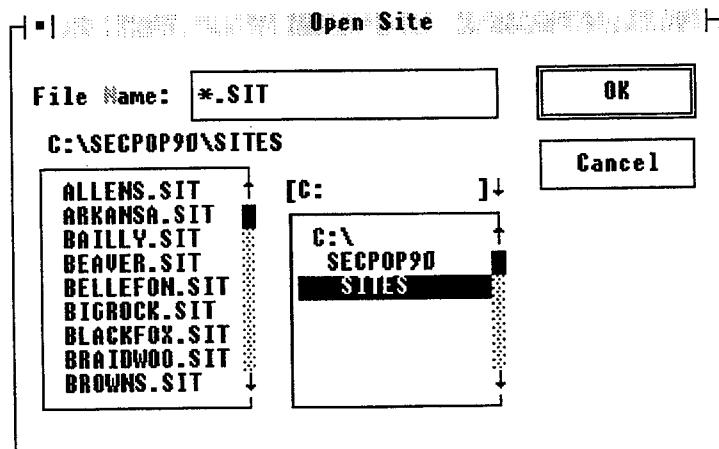


Figure 3.7 Open Site Form

All predefined sites have the default filename extension "SIT." The user is not required to follow this practice. Doing so, however, will make it easier to find previously defined site files. This is especially true when a single directory has a mix of site, problem, and results files.

If either *New Site* or *Open Site* is selected from the *Site* sub-menu and site data have already been entered or loaded from a file, SECPOP90 will first ask the user if the previously entered or loaded data should be used before overwriting it. SECPOP90 displays the following message:

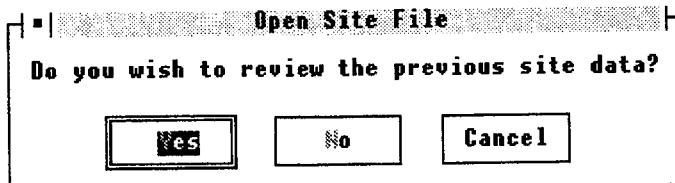


Figure 3.8 Previous Site Data

If the user selects *Yes*, then the *New Site* or *Open Site* operation is canceled and the *Site* form with the old data is displayed. If the user selects *No*, then the old data is overwritten with a blank form or the new data. Selecting *Cancel* cancels the *New Site* or *Open Site* operation and returns the user back to the main menu.

### 3.3.3 Save Site

Selecting *Save Site* saves the site information using the filename displayed in the *Site* form *Title Bar*. After the file is saved, the *Title Bar* of the *Site* form is updated. This command is equivalent to the *Save* command found on the *Site* form (see Figure 3.4).

### 3.3.4 Save As Site

Selecting *Save As Site* opens the form shown in Figure 3.9. This form allows the user to save the site information using a user-specified filename. The form has fields for selecting the appropriate disk drive, directory, sub-directories, and filename. Once these are chosen and the *OK* button is selected, the site-specific information is stored in the specified file. The *Title Bar* of the *Site* form is then updated with the new filename. Selecting *Cancel* cancels the *Save As Site* operation and returns the user back to the main menu. This command is equivalent to the *Save As* command found on the *Site* form (see Figure 3.4).

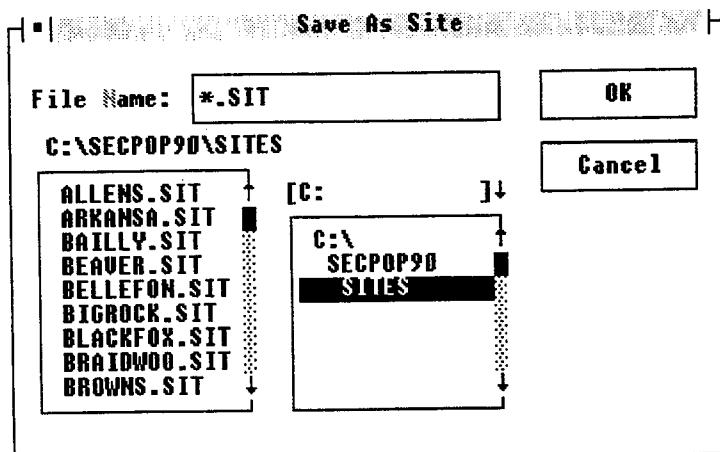


Figure 3.9 Save As Site Form

## 3.4 Problem Definition

Problem definition consists of specifying the radii of interest, the population multiplier, the name of the site file which contains the site-specific information for the problem, mapping the problem sectors to economic regions and any desired problem-specific remarks. The *Problem* sub-menu has four available commands. Note that some of these commands are not available at all times. For example, *Save Problem* and *Save As Problem* are not enabled if there is no problem information to save. The four *Problem* sub-menu commands are

<b>New Problem</b>	enter the problem-specific parameters for a new problem,
<b>Open Problem</b>	open a previously saved problem,
<b>Save Problem</b>	save the problem information with the present filename, and
<b>Save As Problem</b>	save the problem information with a new filename

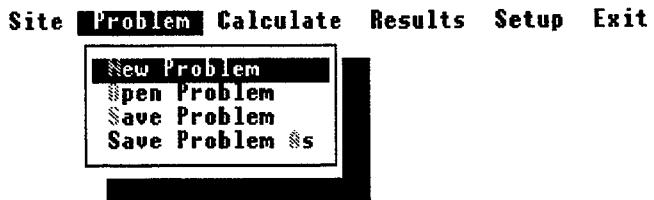


Figure 3.10 Problem Sub-Menu

### 3.4.1 New Problem

Selecting *New Problem* opens the form shown in Figure 3.11. The *Problem* form allows the user to enter or edit all problem-specific information

Figure 3.11 Problem Form

#### Problem Form Fields

The *Problem* form (see Figure 3.11) consists of the following fields which are designed so that the user may easily edit all of the problem-specific information (fields are designated required or optional depending on whether they are required to perform site-specific calculations):

**Title Bar** contains the filename of the problem being edited. If this is a new problem, then the default filename "NEW\_PROB.SIT" will be used. If the *Problem* form is edited, then the phrase "(Modified)" will be appended to the problem filename. The filename is automatically updated when the user performs a *Save* or *Save As* command (see the next section)

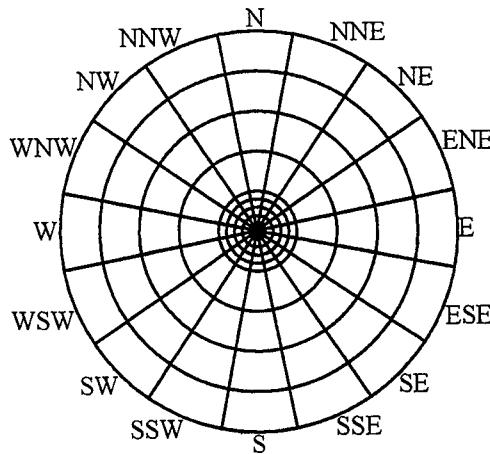
**Site File Name** the name of the site file associated with this problem. This field is required.

**Population Multiplier** a multiplier that is applied to the 1990 census population data. This field allows manipulation of the 1990 census data to estimate changes in the population by a uniform multiplication factor. For example, if it is expected that in the year 2000 the population will increase by 10% in a given location, a population scale factor of 1.1 could be used to represent this growth. The default value is 1.0 — representing the 1990 census population. A value less than or equal to 0.0 is not allowed. This field is required.

**Radial Distances** the radii of interest for the problem. The geometry of SECPOP90 problems is based on that of MACCS. The area around a site is divided into sixteen directions that are equivalent to a standard navigational compass rosette. The rosette is then further divided by radial rings specified by the user using the *Radial Distances* field. Figure 3.12 shows a rosette that has nine user-defined radial rings. As is common in practice, there is a tight collection of "inner" rings surrounded by a looser collection of "outer" rings. The site itself is located at the very center of the rosette.

The collection of directional lines and radii naturally divide the rosette into sections that are defined by two consecutive radii and two neighboring directional lines (except the innermost sections which are defined by the first radius and neighboring directional lines; normally this

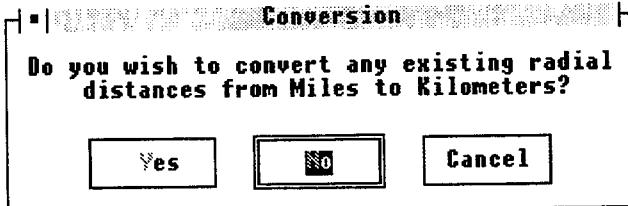
innermost radius is defined to be the exclusion area of the site, the second radius is often defined to be the low population zone) These sections are mapped into economic regions by using the *Edit Regions* command described below



**Figure 3.12 Rosette Definition**

The radial distances can be expressed in miles or kilometers (see the *Radial Units* field described below). The user is limited to thirty-five radii to maintain compatibility with the MACCS Site Data File format. This field is required. At least two radii must be defined. All radii must be separated from each other by at least 0.1 km (0.06214 mi).

**Radial Units, MI** indicates that the radial distances are in miles. If the radial distances are in kilometers and this option is selected, then the user is given the option of performing a conversion from miles to kilometers or only changing the unit designation. This is accomplished using the form shown in Figure 3.13. Either miles or kilometers must be selected.



**Figure 3.13 Radial Unit Conversion**

**Radial Units, KM** indicates that the radial distances are in kilometers. If the radial distances are in miles and this option is selected, then the user is given the option of performing a conversion from kilometers to miles or only changing the unit designation. This is accomplished using a form similar to that shown in Figure 3.13. Either kilometers or miles must be selected.

**Problem Remarks** descriptive remarks about the problem. Up to eighty characters (both rows of the field in the form) may be used to describe the problem. This field will be displayed or printed when viewing this form or printing the problem-specific information (see the *Print* command in the next section). One use of this space could be to record the source of problem-specific information. For example.

**Grand Gulf Nuclear Station (GGNS) Unit: 1 Emergency Plan**

**Section 2.2**

This field is optional

### Problem Form Commands

The *Problem* form has six command buttons: five at the bottom of the *Problem* form similar to the *Site* form and an additional command button for editing the economic regions. The commands are executed when the buttons are selected. The commands support editing of the economic regions and the opening, closing, saving, and printing of the *Problem* form. The command buttons are as follows.

**Edit Regions** opens the *Economic Regions* form (see Figure 3 14) and allows editing of the economic regions. This form is described in detail in section 3 4.2

**Problems** opens a previously defined problem file. This command is equivalent to the *Open Problem* command of the *Problem* sub-menu (see Figure 3 10) For a more detailed explanation, see section 3 4.2

**Close** closes the *Problem* form. All information on the form is saved in memory and can be reviewed by opening the *Problem* form again. The information will be lost unless the problem information is stored in a file (see the *Save* and *Save As* commands below) before SECPOP90 is exited

**Save** saves the problem information using the filename displayed in the *Problem* form *Title Bar*. This command is equivalent to the *Save Problem* command of the *Problem* sub-menu.

**Save As** saves the problem information using a filename specified by the user. This command is equivalent to the *Save As Problem* command of the *Problem* sub-menu. For a more detailed explanation, see section 3 4.5

**Print** prints the problem-specific information to a connected printer or a file. When this button is selected, the form in Figure 3 6 will be displayed

This form allows the user to select which printer or file the problem information is to be printed to. In the case of printing to a file, the user can also select whether to replace an existing file or append this information to the end of it. The user can also specify how many copies of the problem-specific information should be printed to either the selected printer or file.

### 3.4.2 Economic Regions

Selecting *Edit Regions* from the *Problem* opens the form shown in Figure 3 14. The *Economic Regions* form allows the user to assign economic regions to the rosette sections or edit the present assignments

Radial Distances in Miles								Regions 97
Sectors	1	2	3	4	5	10	20	
N	2	2	2	2	3	4		
NNE	8	8	8	8	9	10		
NE	14	14	14	14	15	16		
ENE	20	20	20	20	21	22		
E	26	26	26	26	27	28		
ESE	32	32	32	32	33	34		
SE	38	38	38	38	39	40		

Left
Right
Up
Down
Close

Regions  
97
Radii  
10
Sort
Default
Clear

Figure 3.14 Economic Regions Form

#### Economic Regions Form Fields

The *Economic Regions* form consists of the following fields which are designed so that the user may assign or edit the economic regions. All of the fields on this form are required. Many of the field are filled automatically by SECPOP90 and cannot be edited by the user. Other fields are editable by the user. These fields will be designated as "User input field," in the descriptions below

**Radial Distances** the first row of the economic regions table contains the radial distances in the units specified on the *Problem* form (see Figure 3 11)

**Sectors** the first column of the economic regions table contains the rosette direction descriptors. These descriptors correspond to the sixteen standard directions of a navigational compass

Regions number of economic regions The minimum number of allowed economic regions is 1 The maximum number of allowed regions is 99.

Radii number of radii The minimum number of allowed radii is 2. The maximum number of allowed radii is 35 The radii are defined on the *Problem* form

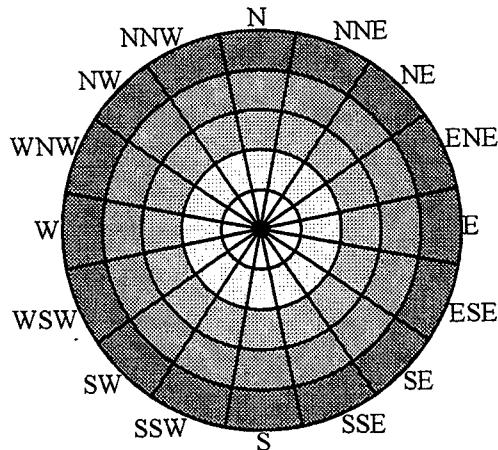
Rosette Sections the remaining values in the economic regions table are the economic region designators for the displayed rosette sections Economic region # 1 is automatically assigned to all of the sections that lie within the first innermost radius This radius is normally thought of as the exclusion area boundary and economic region # 1, the exclusion area

Radial Distances in Miles						Regions
Sectors	5	10	15	20	25	5
N	2	3	4	5		
NNE	2	3	4	5		
NE	2	3	4	5		
ENE	2	3	4	5		
E	2	3	4	5		
ESE	2	3	4	5		
SE	2	3	4	5		

**Sort** **Default** **Clear** **Close**

**Left** **Right** **Up** **Down**

Figure 3.15 Example Economic Regions Form



LEGEND — ECONOMIC REGIONS

EXCLUSION	REGION_02	REGION_03	REGION_04	REGION_05

Figure 3.16 Example Economic Regions Rosette

To assign economic regions to the remaining rosette sections, enter a number for that region that is greater than or equal to 2 and less than or equal to 99 That rosette section then belongs to that economic region If you enter a 2 in the third column, second row of the economic regions table shown in Figure 3.14, then the rosette section defined by the first and second radii in the northern direction would be part of economic region 2 For example, if you wish to define four economic regions in addition to the exclusion area, and you want them to be simple rings centered around the

site, then if you have five radii, your *Economic Regions* should look like Figure 3.15. This corresponds to the economic regions mapping illustrated in Figure 3.16. (Note this figure is for illustrative purposes only and does not represent a SECPOP90 screen. Also, actual economic regions are likely to be much more irregular.) The user is not limited to simple geometries such as that shown. Economic regions can be assigned to any rosette section. There is no need for the same economic region to occupy adjacent cells or for all economic regions to occupy the same number of cells. This allows the user to assign a single economic region to features that might occur repeatedly over the rosette. For example, a user could assign the same economic regions to lakes or desert areas.

### Economic Regions Form Commands

The *Economic Regions* form has eight command buttons: four at the bottom of the *Economic Regions* form that are used for scrolling through economic region values, three command buttons on the right-hand side of the form for performing various operations on the economic regions, and a button for closing the *Economic Regions* form. The commands are executed when the buttons are selected. The command buttons are as follows:

<b>Left</b>	scrolls the economic regions table left so that the economic regions of more distant radii can be edited
<b>Right</b>	scrolls the economic regions table right so that the economic regions of less distant radii can be edited
<b>Up</b>	scrolls the economic regions table up so that the economic regions of more northwestern directions can be edited
<b>Down</b>	scrolls the economic regions table down so that the economic regions of more northeastern directions can be edited
<b>Sort</b>	renumbers economic regions so that there are no gaps in the numbering of the regions and the regions are sorted in row column order. Sorting the economic regions is not required but it normally makes the economic regions mapping to the rosette sections easier to understand.
<b>Default</b>	restores the economic regions to the default values. These values are the same values used if the user does no editing of the economic regions.
<b>Clear</b>	clear all values from all of the rosette sections except for economic region 1, which is always the section from the center of the rosette to the first radius — normally considered to be the exclusion area
<b>Close</b>	closes the <i>Economic Regions</i> form. To save the edited economic regions, the user must select <i>Save</i> or <i>Save As</i> from the <i>Problem</i> form (see Figure 3.11) or <i>Save Problem</i> or <i>Save As Problem</i> from the <i>Problem</i> sub-menu (see Figure 3.10).

### 3.4.3 Open Problem

Selecting *Open Problem* opens the form shown in Figure 3.17. This form allows the user to select a previously defined problem file. The form has fields for selecting the appropriate disk drive, directory, sub-directories, and filename. Once these are chosen and the *OK* button is selected, the file is loaded into memory and the *Problem* form will be displayed with the saved information (see Figure 3.11). The user can close the *Open Problem* form by selecting *Cancel*, pressing ESC, or selecting *Close* from the control box sub-menu.

It is recommended that the user use filename extension “PRB” for all problem files. The user is not required to follow this practice. Doing so, however, will make it easier to find previously defined problem files. This is especially true when a single directory has a mix of site, problem, and results files.

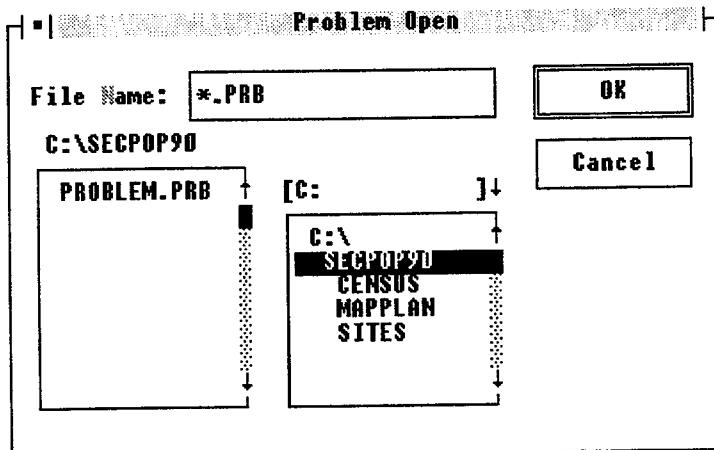


Figure 3.17 Open Problem Form

If either *New Problem* or *Open Problem* are selected from the *Problem* sub-menu and problem data has already been entered or loaded from a file, SECPOP90 will first ask the user if the previously entered or loaded data should be used before overwriting it. SECPOP90 displays the following message

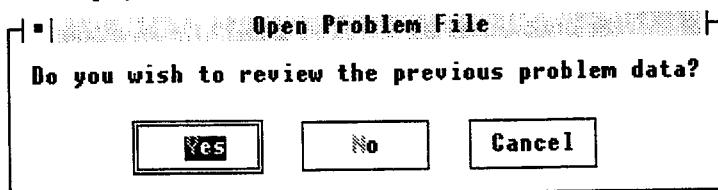


Figure 3.18 Previous Problem Data

If the user selects *Yes*, then the *New Problem* or *Open Problem* operation is canceled and the *Problem* form with the old data is displayed. If the user selects *No*, then the old data are overwritten with a blank form or the new data. Selecting *Cancel* cancels the *New Problem* or *Open Problem* operation and returns the user back to the main menu

#### 3.4.4 Save Problem

Selecting *Save Problem* saves the problem information using the filename displayed in the *Problem* form *Title Bar*. After the file is saved, the *Title Bar* of the *Problem* form is updated. This command is equivalent to the *Save* command found on the *Problem* form (see Figure 3.11)

#### 3.4.5 Save As Problem

Selecting *Save As Problem* opens the form shown in Figure 3.19. This form allows the user to save the problem information using a user-specified filename. The form has fields for selecting the appropriate disk drive, directory, sub-directories, and filename. Once these are chosen and the *OK* button is selected, the problem-specific information is stored in the specified file. The *Title Bar* of the *Problem* form is then updated with the new filename. Selecting *Cancel* cancels the *Save As Problem* operation and returns the user back to the main menu. This command is equivalent to the *Save As* command found on the *Problem* form (see Figure 3.11)

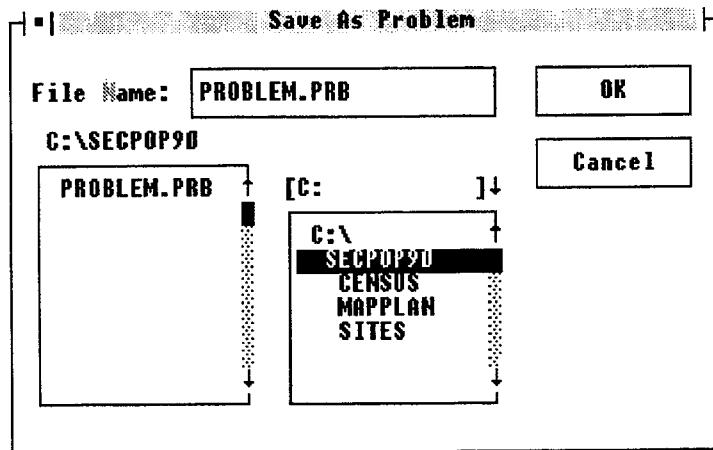


Figure 3.19 Save As Problem Form

### 3.5 Performing Site-Specific Calculations

Once the *Site*, *Problem*, and *Economic Regions* forms are complete, a site-specific calculation may be performed by selecting the *Calculate* sub-menu (see Figure 3.20) from the main menu and then selecting *Site-Specific* from the *Calculate* sub-menu

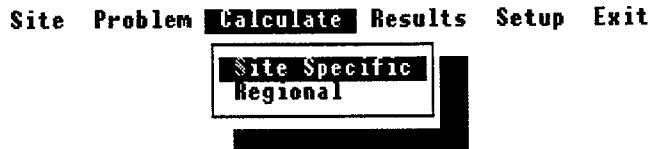


Figure 3.20 Calculate Sub-Menu

If there isn't enough information to perform a site-specific calculation, an error message (see Figure 3.21) is displayed and the user must provide the necessary information before the calculation can proceed

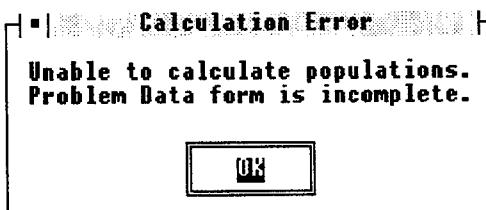


Figure 3.21 Calculation Error

After all of the forms have been sufficiently completed, the *Calculate* form (see Figure 3.22) will be displayed. To begin the site-specific calculation, select the *Calculate* button

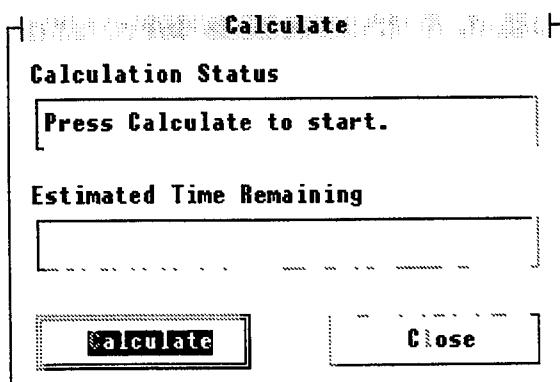


Figure 3.22 Initial Calculate Form

## Calculate Form Fields

The *Calculate* form consists of the following fields which are designed so that the user may monitor the progress of the calculation. All of the fields are filled in automatically by SECPOP90 and cannot be edited by the user.

Calculation Status	the present status of the calculation. Initially this field instructs the user to press the <i>Calculate</i> button.
	Subsequent phases of the calculation display messages describing the phase of the calculation under way (see Figure 3.23)
Estimated Time Remaining	an estimate of how much time remains before the calculation is complete.

## Calculate Form Commands

The *Calculate* form has two command buttons that are used to control the calculation and close the *Calculate* form. The commands are executed when the buttons are selected. The command buttons are as follows:

Calculate	begin the site-specific calculation, and
Close	stop the calculation if a calculation is under way. Note the user is asked to confirm that the calculation should be stopped (see Figure 3.24). If a calculation is not under way, then this command closes the <i>Calculate</i> form.

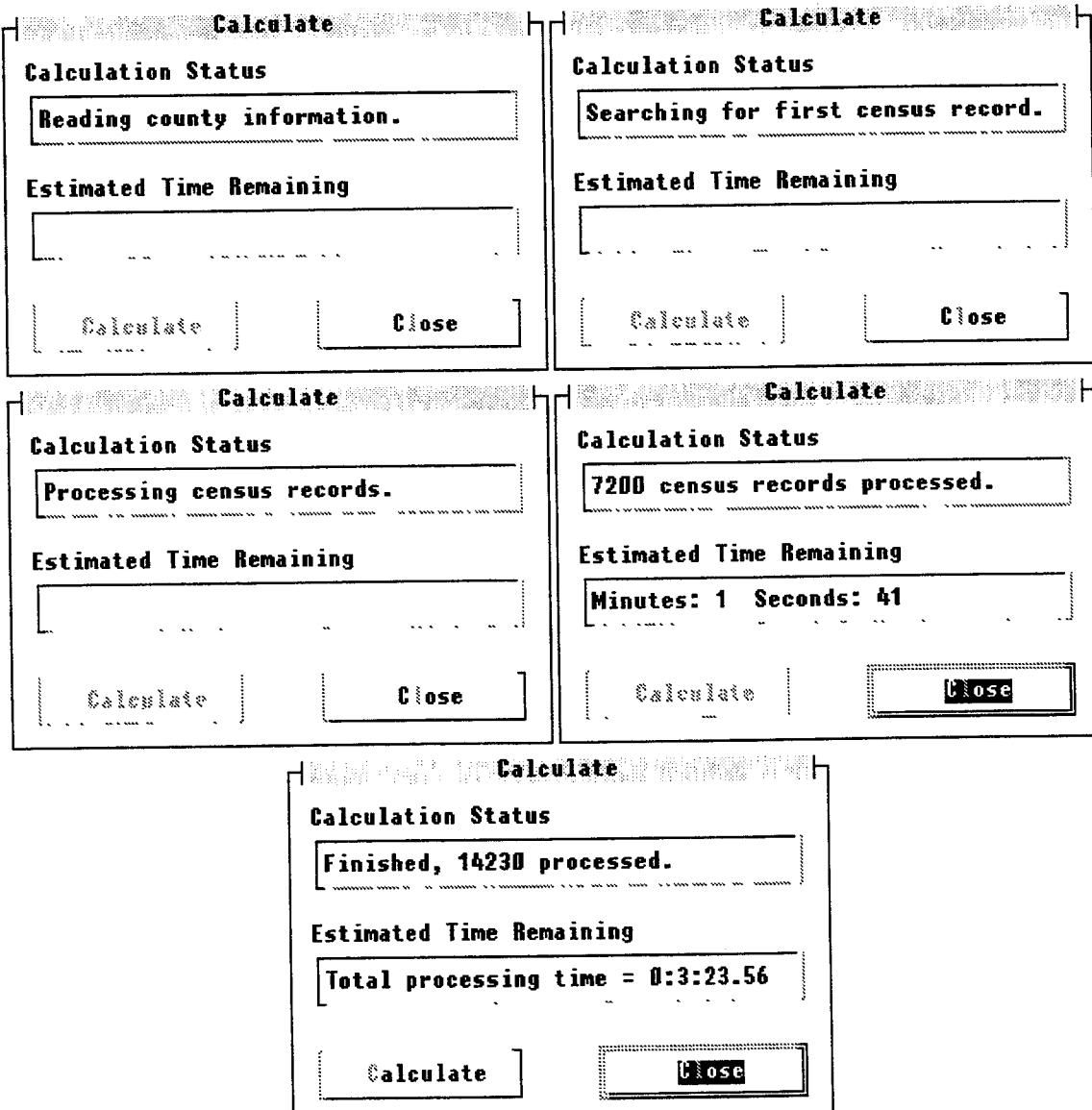


Figure 3.23 Subsequent Calculate Forms

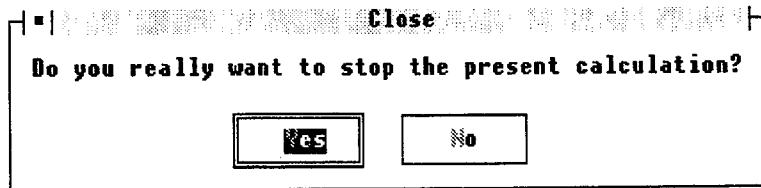


Figure 3.24 Stop Calculation

Note the population estimates are based on over six million 1990 census block records and are as reasonable estimates as can be made for anywhere in the continental United States. The economic factor and land fraction estimates are aggregated at the county level. This makes these estimates coarser estimates than the population estimates. In addition the various notes and exceptions to the county aggregated data should be reviewed (see section 5.2.1 and Appendix C).

## 3.6 Site-Specific Results

There are several ways to display, print, or save the site-specific calculation results. The results are accessed by selecting the *Results* sub-menu from the main menu (see Figure 3.25). All of the commands and sub-sub-menus except for *Regional* apply to site-specific results and will be discussed below. *Regional* which applies to regional calculation, is discussed in section 3.7.

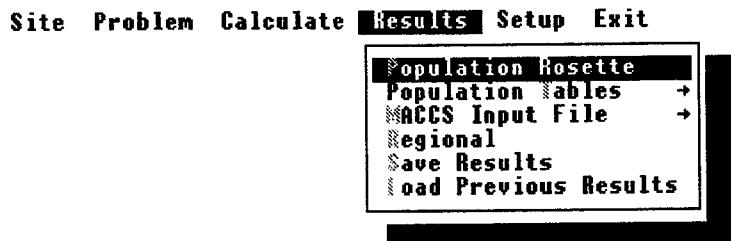


Figure 3.25 Results Sub-Menu

Population Rosette	displays a rosette with populations per section,
Population Tables	displays rosette populations in table formats,
MACCS Input File	displays or prints the MACCS Site Data File;
Save Results	saves the calculated results, and
Load Previous Results	loads previously saved calculated results

### 3.6.1 Population Rosette

Selecting *Population Rosette* from the *Results* sub-menu displays a rosette as shown in Figure 3.26. This is a graphics screen. There are no fields or commands since it is not a form. Pressing SPACE returns the user to the main menu. To save or print the rosette, see section 2.1.

### 3.6.2 Population Tables

Selecting *Population Tables* from the *Results* sub-menu displays the *Population Tables* sub-sub-menu (see Figure 3.27). This menu provides the user with two methods for displaying the population results, the *Population* command which displays a table version of the rosette with the population values for each rosette section, and the *Cumulative* command which displays a similar table except that the population sums along each direction are displayed.

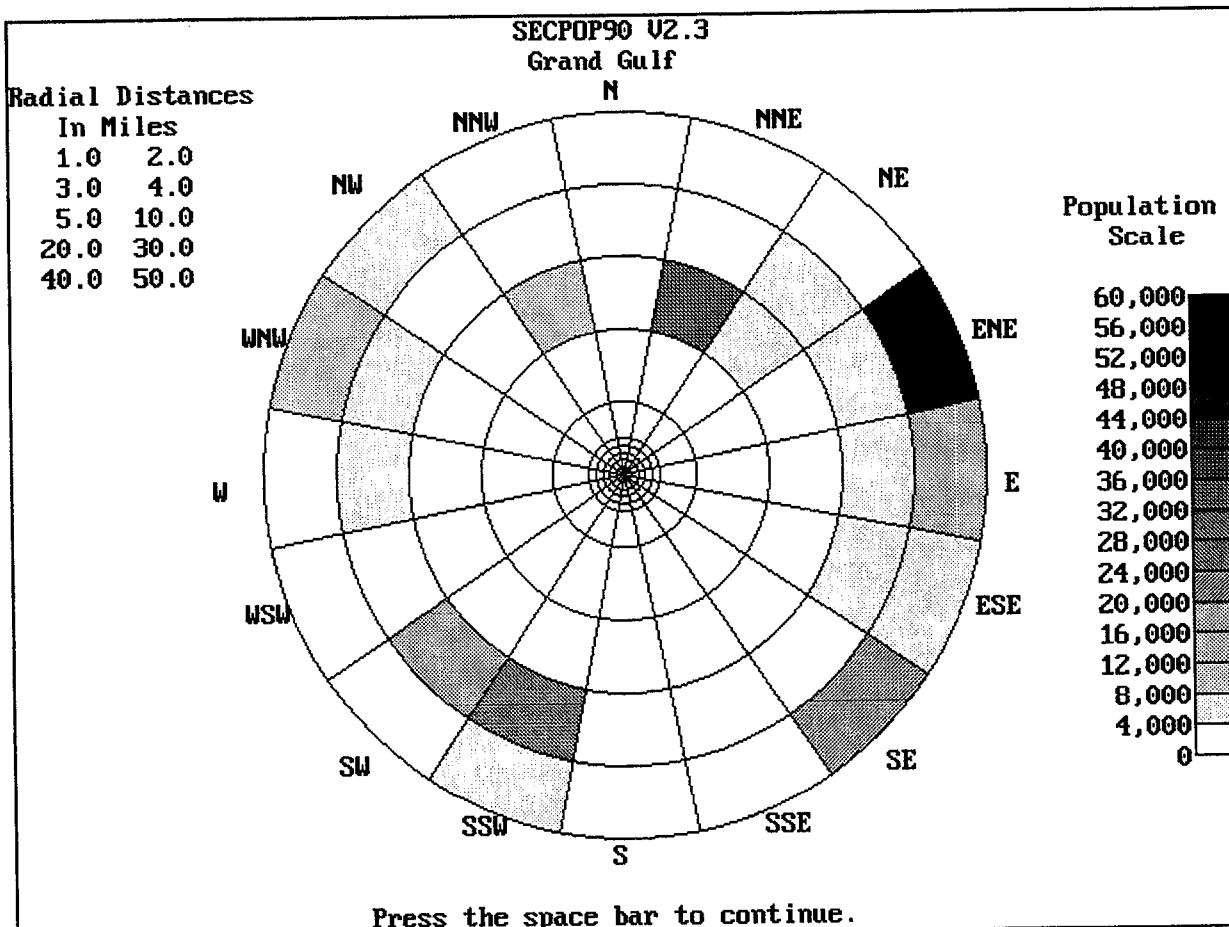


Figure 3.26 Population Rosette

Site Problem Calculate **Results** Setup Exit

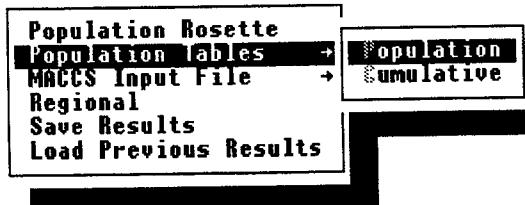


Figure 3.27 Population Tables Sub-Sub-Menu

### Population

Selecting *Population* from the *Population Tables* sub-sub-menu displays the table shown in Figure 3.28. The population data are shown in a manner similar to the method used to display the economic regions on the *Economic Regions* form (see section 3.4.2). The table has no user input fields. There are two command buttons, *Close* and *Print*, that perform the following functions:

<b>Close</b>	closes the <i>Population Table</i> form, and
<b>Print</b>	print the population information to the selected printer or file. Selecting this command displays a <i>Print</i> form similar to Figure 3.6.

### Cumulative

Selecting *Cumulative* from the *Population Tables* sub-sub-menu displays the table shown in Figure 3.29. The cumulative population data are shown in a manner similar to the method used to display the economic regions on the *Economic*

*Regions* form (see section 3.4.2). The population is accumulated along each direction. This means that along each direction the radial population includes all of the population up to and including that radius for the given direction

Population Table					
	Radii (Miles)				
	1.0000	2.0000	3.0000	4.0000	Sum
N	0	21	0	0	2457
NNE	0	0	0	0	40758
NE	0	15	0	1	13649
ENE	0	36	0	74	67873
E	2	0	8	80	19663
ESE	0	21	0	5	15621
Sum	2	103	18	228	318174

Figure 3.28 Population Table Form

For example, in Figure 3.29 the cumulative population at the 4.0-mile radius along the eastern direction is 90. This value includes the population of 2 found within the 1.0-mile radius, the population of 8 between the 2.0 and 3.0-mile radii, and the population of 80 found between the 3.0 and 4.0 radii (see also Figure 3.28). The table has no user input fields. There are two command buttons, *Close* and *Print*, that perform the following functions

Close	closes the <i>Cumulative Population Table</i> form, and
Print	prints the cumulative population information to the selected printer or file. Selecting this command displays a <i>Print</i> form similar to Figure 3.6.

Cumulative (By Direction) Population Table					
	Radii (Miles)				
	1.0000	2.0000	3.0000	4.0000	Sum
N	0	21	21	21	2457
NNE	0	0	0	0	40758
NE	0	15	15	16	13649
ENE	0	36	36	110	67873
E	2	2	10	90	19663
ESE	0	21	21	26	15621
Sum	2	105	123	351	318174

Figure 3.29 Cumulative Population Table Form

### 3.6.3 MACCS Input File

Selecting *MACCS Input File* from the *Result* sub-menu displays the *MACCS Input File* sub-sub-menu (see Figure 3.30). This menu provides the user with two methods for displaying the population results, the *Display* command which displays the MACCS Site Data File, and the *Print* command which prints the MACCS Site Data File to the selected printer or file

#### Display

Selecting *Display* from the *MACCS Input File* sub-sub-menu displays the *MACCS Site File* form shown in Figure 3.31. There are no user input fields on this form. The user can scroll through the data by selecting the arrows. The command *Close* closes the form. The MACCS Input File is normally displayed as ASCII text. If the user has selected the Comma-

Separated Variables (CSV) option in the *Setup* form (see section 3.9.1), then the data will be displayed in CSV format. All text strings will be enclosed in double quotes and all variables (including text strings) will be separated by commas. This format is compatible with the import data feature of most spreadsheet programs.

Site Problem Calculate Results Setup Exit

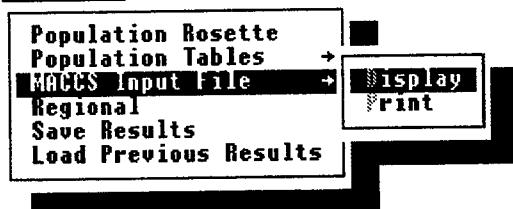


Figure 3.30 MACCS Input File Sub-Sub-Menu

SECPOP90 MACCS Site Data File

SECPOP90 V2.3 MACCS Site Data File for Grand Gulf							
Lat: 32° 0'27" Long: 91° 2'53" Population multiplier: 1.0000 09							
10 SPATIAL INTERVALS							
16 WIND DIRECTIONS							
1 CROP CATEGORIES							
1 WATER PATHWAY ISOTOPES							
1 WATERSHEDS							
97 ECONOMIC REGIONS							
SPATIAL DISTANCES		KILOMETERS					
1.61	3.22	4.83	6.44	8.05	16.09	32.19	
64.37	80.47						
POPULATION							
0.	21.	0.	0.	0.	0.	137.	
829.	841.						
<b>Close</b>							

Figure 3.31 MACCS Site Data File Form

### Print

Selecting *Print* from the *MACCS Input File* sub-sub-menu prints the MACCS Site Data File to the selected printer or file. Selecting this command displays a Print form similar to Figure 3.6. The MACCS Input File is normally printed as ASCII text. If the user has selected the Comma-Separated Variables (CSV) option in the *Setup* form (see section 3.9.1) then the data will be printed in CSV format. The following is an example MACCS Site Data File. Note SECPOP90 does not perform any calculations for estimating crop categories, water pathway isotopes, or watersheds. The values below for those categories (watershed index, crop season and share, and watershed definition) are dummy values generated so that the file created will be completely compatible with MACCS. For a full explanation of all of the data fields, see Appendix B.

Table 3.1 Example MACCS Site Data File

SECPOP90 V2.3 MACCS Site Data File for Grand Gulf							
Lat: 32° 0'27" Long: 91° 2'53" Population multiplier: 1.0000 07-17-1997							
10 SPATIAL INTERVALS							
16 WIND DIRECTIONS							
1 CROP CATEGORIES							
1 WATER PATHWAY ISOTOPES							
1 WATERSHEDS							
97 ECONOMIC REGIONS							
SPATIAL DISTANCES		KILOMETERS					
1.61	3.22	4.83	6.44	8.05	16.09	32.19	48.28
64.37	80.47						
POPULATION							
0.	21.	0.	0.	0.	0.	137.	629.
829.	841.	0.	0.	19	0.	3985.	32472
3481.	801.	0.	0.	36	47.	1635.	4666.
4532.	2717.	0.	1.	36	47.	1635.	4666.
0.	36.	0.	74.	0.	128.	264.	3667.
4018.	59686.						

2.	0.	8	80.	54	144.	513.	1421.
4029.	13412						
0.	21.	0	5	327.	765.	676.	275.
5688.	7864						
0.	0	0.	15.	747.	2607.	1088.	491.
2143	20590.						
0.	4.	8.	51.	82.	533.	977.	352.
3167.	2548.						
0.	0.	0	0.	0.	64.	2124.	3808.
1921.	1276						
0.	1	2.	2.	0	54.	1621.	2811.
26191.	4314						
0.	0.	0.	0.	0.	35.	1204.	1513.
17862.	3273.						
0.	0.	0.	0.	0	22.	1784.	469.
1859.	1526.						
0.	0.	0.	0.	0.	268.	357.	279.
4638.	3141						
0.	0.	0.	0.	0.	40.	2029.	275.
4610.	11505.						
0.	0.	0.	0.	0.	12.	177.	46.
2303.	6361.						
0.	5.	0.	0.	0.	1.	96.	9876.
699.	2296.						

### LAND FRACTION

LAND FRACTION	0.97	0.97	0.00	0.00	0.00	0.96	0.96	0.96	0.95	0.95
	0.00	0.00	0.00	0.00	0.97	0.95	0.95	0.95	0.94	0.96
	0.00	0.97	0.00	0.97	0.97	0.96	0.95	0.96	0.97	0.98
	0.97	0.97	0.00	0.97	0.00	0.97	0.97	0.99	0.99	0.99
	0.97	0.00	0.97	0.97	0.97	0.97	0.97	1.00	0.99	0.99
	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.99	1.00	1.00
	0.00	0.00	0.00	0.97	0.97	0.97	0.97	0.99	1.00	1.00
	0.97	0.97	0.97	0.97	0.97	0.97	0.98	0.99	1.00	1.00
	0.97	0.00	0.00	0.97	0.97	0.97	0.99	0.99	0.99	0.99
	0.97	0.97	0.97	0.97	0.97	0.97	0.98	0.97	0.95	0.95
	0.00	0.00	0.00	0.00	0.97	0.97	0.98	0.96	0.93	0.93
	0.00	0.00	0.97	0.00	0.94	0.97	0.94	0.94	0.95	0.95
	0.00	0.97	0.00	0.94	0.94	0.94	0.94	0.95	0.98	0.97
	0.00	0.00	0.00	0.94	0.94	0.95	0.94	0.94	0.98	0.98
	0.00	0.97	0.97	0.95	0.94	0.94	0.94	0.94	0.96	0.97
	0.00	0.97	0.00	0.00	0.96	0.94	0.95	0.96	0.96	0.96

### REGION INDEX

REGION	INDEX
1	2 2 2 2 2 3 4 5 6 7
1	8 8 8 8 9 10 11 12 13
11	14 14 14 15 16 17 18 19
12	20 20 20 21 22 23 24 25
12	26 26 26 27 28 29 30 31
13	23 23 23 23 33 34 35 36 37
13	88 88 88 94 04 14 24 43
14	44 44 44 44 54 64 74 84 93
15	05 05 05 05 15 25 35 45 55
15	65 65 65 65 75 85 95 60 61
16	26 26 26 26 36 46 56 66 77
16	86 86 86 86 97 07 17 27 33
17	47 47 47 47 57 67 77 87 99
18	08 08 08 08 18 28 38 48 58
18	68 68 68 68 78 88 99 09 11
19	29 29 29 29 39 49 59 69 79

## WATERSHED INDEX

#### **CROP SEASON AND SHARE**

1. NONE 1. 1. 0.0000

WATERSHED DEFINITION		0.0	0.0	0.0	0 0
1	NONE				
REGIONAL ECONOMIC DATA					
1	EXCLUSION	0.284 .015	225.	1594.	67075.
2	REGION_02	0.284 .015	225	1594.	67075.
3	REGION_03	0.430 .006	388.	1871.	74951.
:	:	:	:	:	:
95	REGION_95	0.617 .000	550.	1948	62374.
96	REGION_96	0.622 .000	556.	1980.	62954.
97	REGION_97	0.689 .003	643.	2433.	72112.

### 3.6.4 Save Results

Selecting *Save Results* from the *Results* sub-menu displays the form shown in Figure 3.32. This form allows the user to save the calculation results using a user-specified filename. The form has fields for selecting the appropriate disk drive, directory, sub-directories, and filename. Once these are chosen and the OK button is selected, the calculation results are stored in the specified file. It is recommended that the user save all result files with the filename extension “.TXT”. The user is not required to follow this practice. Doing so, however, will make it easier to find previously saved calculation results files. This is especially true when a single directory has a mix of site, problem, and results files. It will also allow third-party software to recognize the files as ASCII text files. This also holds true if the files are saved as CSV files (see section 3.6.3).

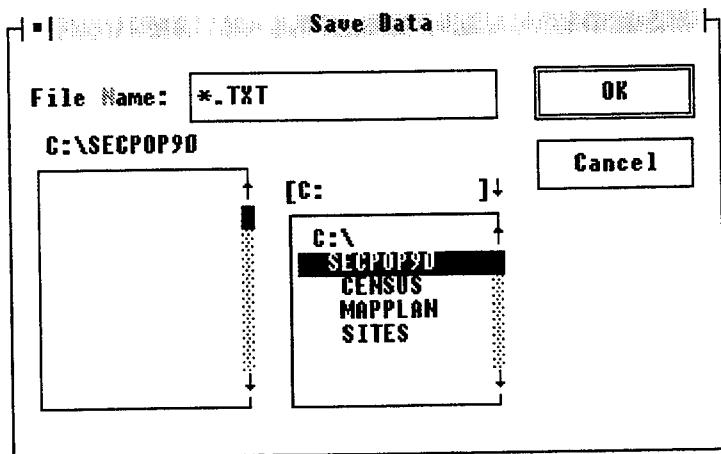


Figure 3.32 Save Data Form

### 3.6.5 Load Previous Results

Selecting *Load Previous Results* from the *Results* sub-menu displays the form shown in Figure 3.33. This form allows the user to select previously calculated results. The form has fields for selecting the appropriate disk drive, directory, sub-directories, and filename. Once these are chosen and the OK button is selected, the file is loaded into memory and the results can be examined using any of the methods described above.

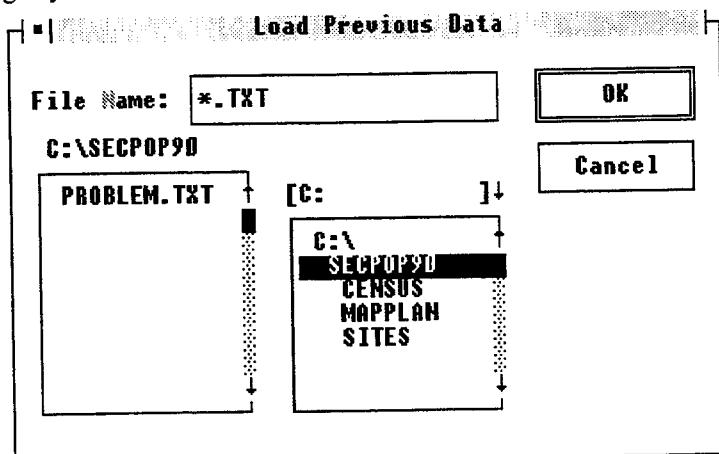


Figure 3.33 Load Previous Results Form

Note: only results saved as ASCII text can be retrieved. Results saved in CSV format cannot be retrieved by SECPOP90. Also, since the results are saved in a format compatible with MACCS, the radial distances will be in kilometers. The radial distances can be changed back to miles by loading in the problem file for the results file after the results are loaded or opening the problem form after the results are loaded and changing the *Radial Units* to miles (see section 3.4.1). If the second method is employed, the user should review the radial distances and edit the values, if needed, to compensate for round off error. The user can close the Open Problem form by selecting *Cancel*, pressing ESC, or selecting *Close* from the control box sub-menu.

### 3.7 Performing Regional Calculations

A regional calculation may be performed by selecting the *Calculate* sub-menu (see Figure 3.34) from the main menu then selecting *Regional* from the *Calculate* sub-menu. Important note: in order for the regional calculations to be displayed and printed correctly, the *MapPlan* command from the *Setup* form (see section 3.9.2) must be run before the first regional calculation. After the *MapPlan* settings have been saved, it is no longer necessary to run the *MapPlan* setup command unless the computer's hardware configuration changes.

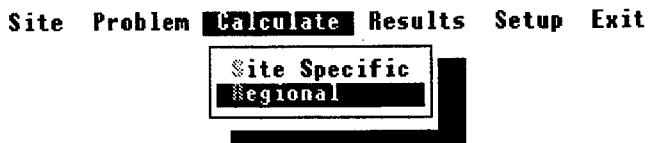


Figure 3.34 Calculate Sub-Menu

Selecting *Regional* opens the form shown in Figure 3.35. The *Regional Calculation Setup* form allows the user to enter all of the information necessary to perform regional calculations.

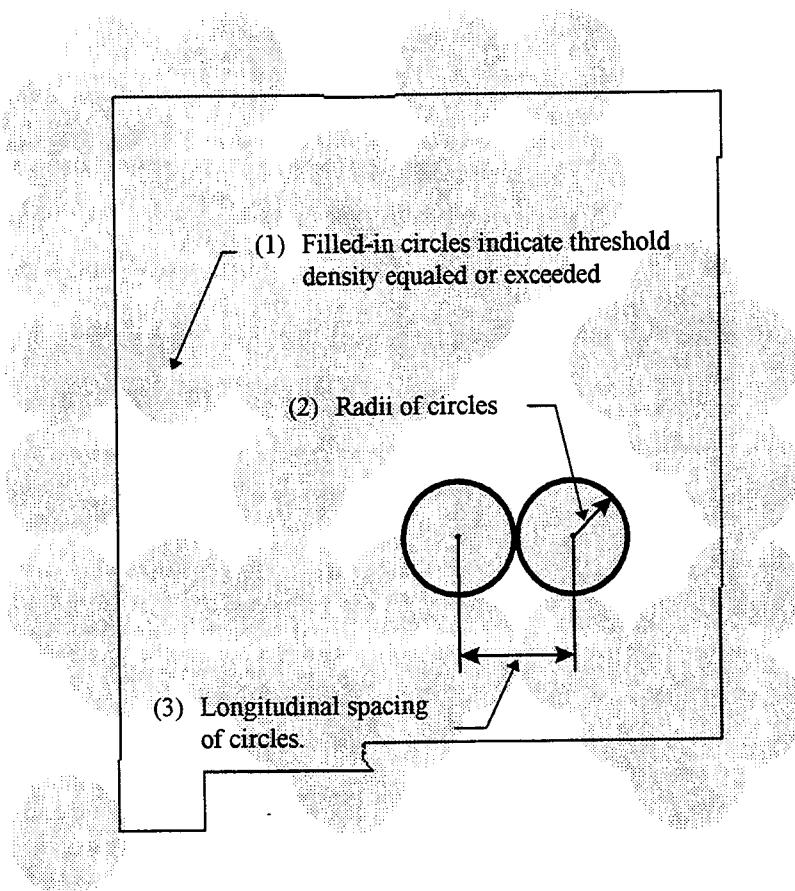
Figure 3.35 Regional Calculation Setup Form

#### Regional Calculation Setup Form Fields

The *Regional Calculation Setup* form (see Figure 3.35) consists of the following fields which are designed so that the user may easily enter all of the regional-calculation specific information (all of the fields are required to perform regional calculations).

**Radii of Circles** the outermost radii of the circles in miles [see (2) in Figure 3.36] used in the regional population density calculations. Population densities are calculated within each circle in the following manner: population densities are calculated for all persons located within 1, 2, 3, . . . 20, 30, 40, . . . 170 miles of the center of each circle. (The last radii will be equal to *Radii of Circles* which has a minimum value of 1 and a maximum value of 170.) If any one of these population densities equal or exceed the *Population Density Threshold*, then a circle whose radii is equal to *Radii of Circles* will be drawn on the output map (see Figure 3.40).

Longitudinal Spacing of Circles	the longitudinal spacing of the circles in miles [see (3) in Figure 3.36] The latitudinal spacing is half the longitudinal spacing and every other row of circles is offset by half of the longitudinal spacing. If the longitudinal spacing is set to twice the radii, then the region will be completely covered
Population Density Threshold	the population density threshold in people per square mile that when equal or exceeded will cause the circle in which the density was equaled or exceeded to be drawn on the output map [see (1) in Figure 3.36 and Figure 3.40].
Input Map Name	the name of the map from the secpop90\mapplan\maps directory (see Figure 2.1). There are maps for each state and its surrounding states (the format of the names are <i>nn_PLUS</i> , where <i>nn</i> is the state's postal abbreviation), the four NRC regions (the format of the names are <i>NRC_REGn</i> , where <i>n</i> is from 1 to 4), and a map of the continental United States (the format of the name is <i>CONUS</i> ).
Display Results	select <i>Circles</i> to draw circles on the output map (see Figure 3.40) or <i>Points</i> to draw only points at the center of all circles that equal or exceed the population density threshold



**Figure 3.36 Radii and Longitudinal Spacing**

#### Regional Calculation Setup Form Commands

The *Regional Calculation Setup* form has two command buttons. Both are located at the bottom of the form and are used to perform the regional calculations or close the *Regional Calculation Setup* form. The command buttons are as follows.

Calculate	begins the regional calculations and displays the <i>Regional Calculation Status</i> form (see Figure 3.37); and
Close	closes the <i>Regional Calculation Setup</i> form

### Regional Calculation Status Form

There are two *Regional Calculation Status* forms. The first form displays the percent of the total number of circles calculated (see Figure 3.37). The second form displays the total time (hh mm ss ss) for the calculations to complete (see Figure 3.38). This second form is displayed at the end of the calculations.

**Important note: depending on the size of the circles and the size of the input map, calculations can last from minutes to several days! Try not to use very small circles with very large regions.**

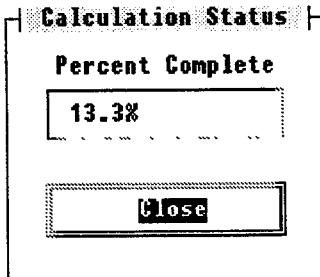


Figure 3.37 Regional Calculation Percent Complete

The command buttons on the *Regional Calculation Status* form control the calculation and close the *Regional Calculation Status* form. If the calculation is not complete and the user selects the *Close* button then the user is asked to confirm that he or she wished to stop the calculation (see Figure 3.24). If confirmation is received, the calculation stops, otherwise, it continues. If the calculation has completed and the user selects the *Continue* button; then the output map is displayed (see Figure 3.40).

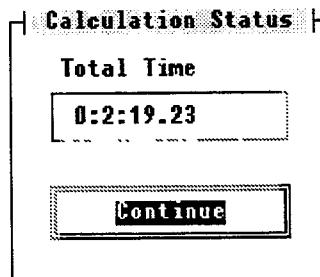


Figure 3.38 Regional Calculation Total Time

## 3.8 Regional Calculation Results

Selecting *Continue* from the *Regional Calculation Status* form or *Regional* from the *Results* sub-menu (see Figure 3.39) displays the *Map Output* form (see Figure 3.40). This is a MapPlan form which displays the most recent regional calculation results. The selected input map is displayed overlaid with the circles in which the population density threshold specified in the *Regional Calculation Setup* form (see Figure 3.35) was equaled or exceeded.

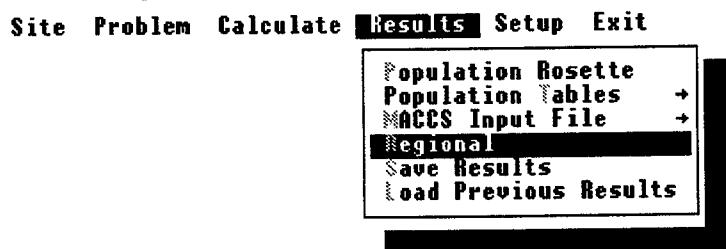


Figure 3.39 Results Sub-Menu

At the top of the output map display is the MapPlan menu bar that operates in a manner similar to the main menu of SECPOP90. For the purposes of SECPOP90, only the *File* sub-menu of the MapPlan menu bar will be discussed. Other sub-menus can be accessed and help can be obtained by highlighting the desired selection and pressing the F1 key. The user is warned, however, that the version of MapPlan which is distributed with SECPOP90 lacks the features necessary to save output files that have been changed.

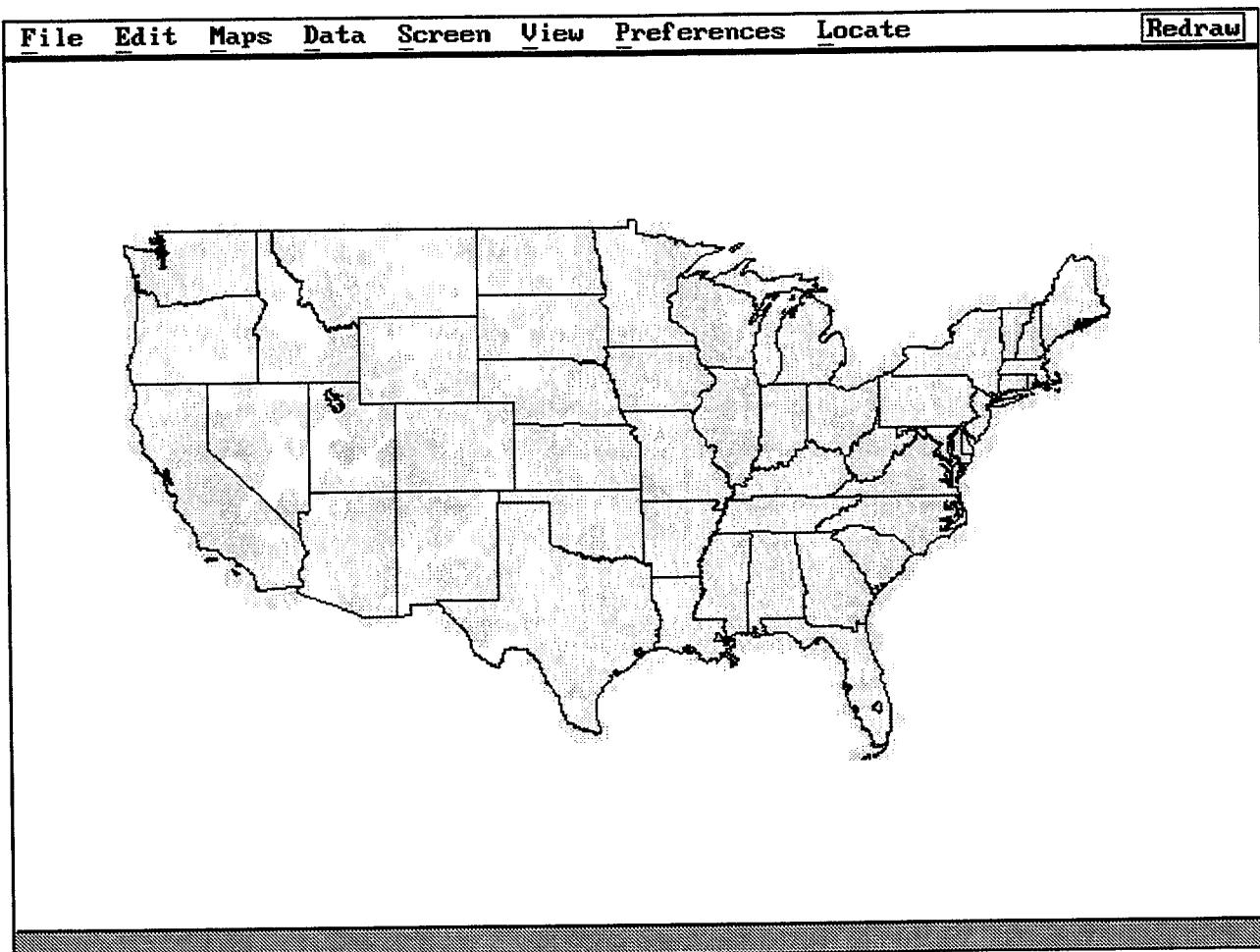


Figure 3.40 Output Map

#### MapPlan File Sub-Menu

The MapPlan *File* sub-menu (see Figure 3.41) has four available commands. The commands are as follows

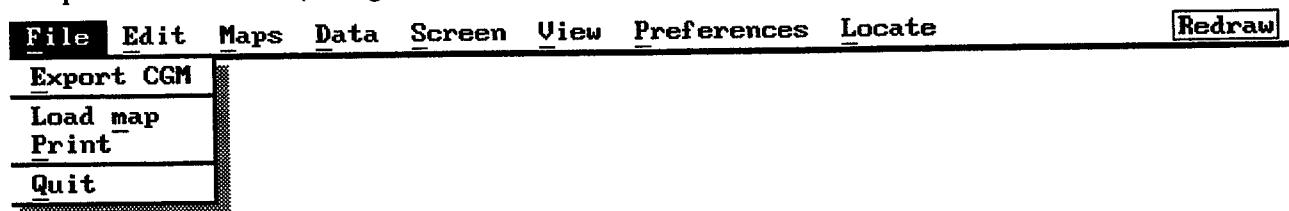


Figure 3.41 MapPlan File Sub-Menu

**Export CGM** exports the output map in Computer Graphics Metafile (CGM) format. The CGM file can then be imported or inserted by other graphics or presentation software. When selected, the *Export CGM File* form appears (see Figure 3.42) and prompts the user for the filename. It is recommended that the complete pathname be given. If only the filename is given then, the file will be written into the *secpop90\mapplan* directory. It is important to note that because of the limitations of this version of MapPlan, the map output will be overwritten the next time a regional calculation is executed. The only way to save the output is by exporting it as explained above or saving the screen image as discussed in section 2.1.

**Load map** loads a previously defined MapPlan map. The maps can be found in the *secpop90\mapplan\maps* sub-directory. The maps available are described in section 3.7. The maps are selected by a form similar in function to Figure 3.33. After the map is loaded, it can be exported as a CGM file or printed.

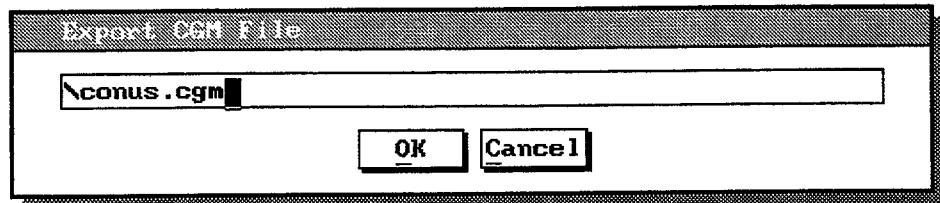


Figure 3.42 Export CGM File Form

**Print** prints the present output map to the printer. Selecting this command opens the *Print Current Image* form. The current device is the selected during the MapPlan setup process (see section 3.9.2). The *Print Current Image* form allows the user to select the paper size — legal or letter, print orientation — portrait or landscape; print resolution — low, medium, or high, the number of copies, and, if the printer supports it, color printing.

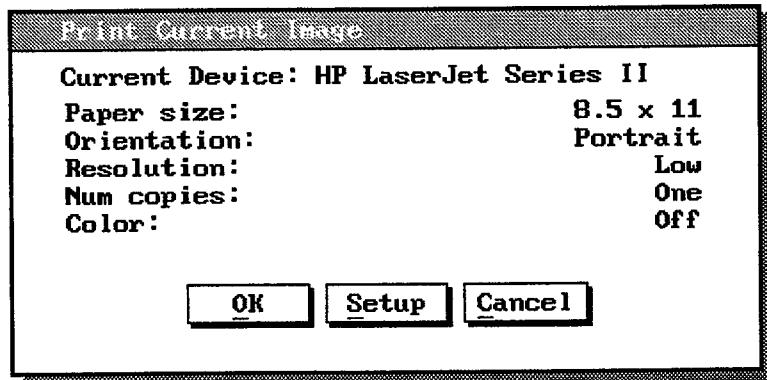


Figure 3.43 Print Current Image Form

Selecting *OK* starts the print job. Selecting *Cancel* cancels the print job if it has not started (there is no way to cancel a print job in progress from MapPlan). Selecting *Setup* displays the *Print Setup* form (see Figure 3.44). The *Print Setup* form allows the user to select the printer output port parameters.

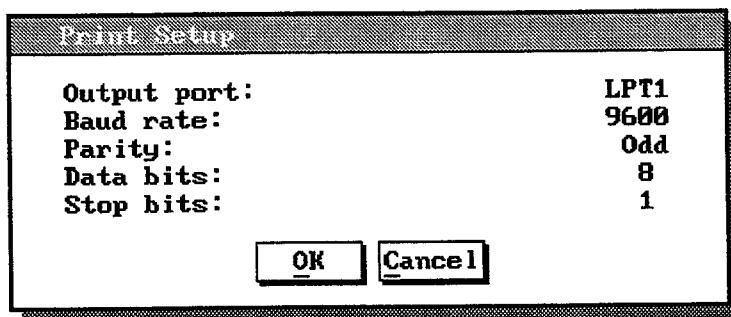


Figure 3.44 Print Setup Form

**Quit** exits the regional calculation output map display and returns to the SECPOP90 main screen.

### 3.9 Customizing SECPOP90

Selecting *Setup* from the main menu displays the *Setup* form shown in Figure 3.45. This form allows the user to customize various SECPOP90 file paths and the type of site-specific calculation results output that will be generated.

Setup	
<b>Default Site File Path</b>	<b>Default Problem File Path</b>
SITES	SITES
<b>Default Output File Path</b>	<b>Location of Census Databases</b>
-	CENSUS
<b>Site-Specific Output File Format</b>	
<input checked="" type="radio"/> MACCS <input type="radio"/> CSV	
<b>MapPlan</b>	<b>Save Changes</b>
<b>Close</b>	

Figure 3.45 Setup Form

### 3.9.1 Setup Form Fields

The *Setup* form has six fields designed to allow the user to specify where all of the important input and output files are to reside. The user can override many of these values using the *File Open* and *Save As* forms discussed earlier but, these pathnames will normally be set to the paths that the user uses most often. For more information about pathnames, see your operating system manual

<b>Default Site File Path</b>	the default path where site files are located. Relative pathnames start at the directory where SECPOP90 is being executed
<b>Default Problem File Path</b>	the default path where problem files are located. Relative pathnames start at the directory where SECPOP90 is being executed
<b>Default Output File Path</b>	the default path where output files will be saved. Relative pathnames start at the directory where SECPOP90 is being executed
<b>Location of Census Databases</b>	the default path where the block-level population and county-level land fraction and economic census databases are located. Relative pathnames start at the directory where SECPOP90 is being executed
<b>Output File Format</b>	select MACCS to save the site-specific calculation results in MACCS Site Data File compatible format. Select CSV to save the results in comma-separated variable format. This format encloses all text strings within double quotes and separates all variables (including text strings) with commas. Both file formats are ASCII text. The differences between these two formats are illustrated by Table 3.2.

### 3.9.2 Setup Form Commands

The *Setup* form has three command buttons one for executing the *MapPlan* setup program, one for saving the user-defined defaults, and the other to close the *Setup* form

**MapPlan** executes the *MapPlan* setup program for specifying the graphics mode, printer type, and printer output options for the *MapPlan* regional calculation output. This program does not support the mouse. Selections are made using the letters g, p, o, and e, along with the arrow keys and the return key. ESC may be pressed at any time to cancel the last operation. Pressing the letter g allows the user to select the graphic display adapter. Pressing the letter p allows the user to select the output printer or plotter. Pressing the letter o allows the user to set up the output printer port. Pressing the letter e saves the *MapPlan* setup information and returns the user back to the SECPOP90 main screen

**Save Changes** immediately changes the default paths, file locations, and file formats. It saves the changes to a special SECPOP90 configuration file. This file will be read the next time SECPOP90 is executed and the default values that have been set by the user will be restored

**Close** closes the *Setup* form. Changes made will not be saved to the SECPOP90 configuration file but will be in effect for the rest of the present SECPOP90 session.

**Table 3.2 MACCS and CSV Formats**

**MACCS Format:**

```
SECPOP90 V2.3 MACCS Site Data File for Grand Gulf
Lat: 32° 0'27" Long: 91° 2'53" Population multiplier: 1.0000 07-17-1997
10 SPATIAL INTERVALS
16 WIND DIRECTIONS
1 CROP CATEGORIES
1 WATER PATHWAY ISOTOPES
1 WATERSHEDS
97 ECONOMIC REGIONS
SPATIAL DISTANCES KILOMETERS
1.61 3.22 4.83 6.44 8.05 16.09 32.19 48.28
64.37 80.47
```

**CSV Format:**

```
"SECPOP90 V2.3 CSV Site Data File for","Grand Gulf"
"Lat:",32,"°", 0,"'",27,"'","Long:", 91,"°", 2,"'",53,"'","Population multiplier:", 1.0000,"07-
17-1997"
10,"SPATIAL INTERVALS"
16,"WIND DIRECTIONS"
1,"CROP CATEGORIES"
1,"WATER PATHWAY ISOTOPES"
1,"WATERSHEDS"
97,"ECONOMIC REGIONS"
SPATIAL DISTANCES KILOMETERS"
1.61, 3.22, 4.83, 6.44, 8.05, 16.09, 32.19, 48.28,
64.37, 80.47,
```

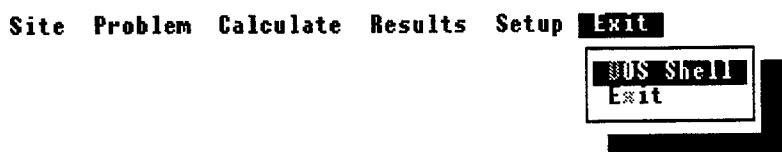
<b>MapPlan</b> <b>Copyright 1990, 1991</b>	<b>Wordtech Systems, Inc.</b>	<b>Version 2.0</b>
<b>GRAPHICS CARD</b> <b>IBM Video Graphics Array (VGA) 640 x 480 (16 Colors)</b>		
<b>PRINTER: LASER PRINTER</b> <b>HP LaserJet Series II</b> <b>PRINTER OPTIONS</b> <b>PARALLEL PORT:1</b>		

**Change Graphics Card, Printer, Options, or Exit (G/P/O/E) ?**  
**Press [Esc] to Quit SETUP without saving changes.**

**Figure 3.46 MapPlan Setup Form**

### 3.10 Exiting SECPOP90

Selecting *Exit* from the main menu displays the *Exit* sub-menu (see Figure 3.47). Two commands are available in the *Exit* sub-menu, *DOS Shell* and *Exit*. These commands will take the user back to the DOS shell either temporarily or permanently.



**Figure 3.47 Exit Sub-Menu**

<b>DOS Shell</b> <b>Exit</b>	execute a temporary DOS shell, and quit the SECPOP90 program
---------------------------------	---

### 3.10.1 DOS Shell

Selecting *DOS Shell* executes a DOS shell, temporarily suspending SECPOP90 and gives the user a DOS prompt (see Figure 3.48) Select this if you wish to do something in DOS but you don't want to stop the program and lose all your information This is useful for doing such things as printing files, making directories, copying files, etc Type **exit** from the DOS prompt to return to SECPOP90 Care should be taken before executing this command if you are not running from DOS Consult your operating system manual

```
Type EXIT to return to SECPOP90
Microsoft(R) MS-DOS(R) Version 6.22
(C)Copyright Microsoft Corp 1981-1994.
C:\SECPOP90>_
```

Figure 3.48 The DOS Shell

### 3.10.2 Exit

Selecting *Exit* will exit the SECPOP90 program The user will be asked to confirm whether the SECPOP90 program should be exited (see Figure 3.49) Exiting SECPOP90 will cause all unsaved information to be lost

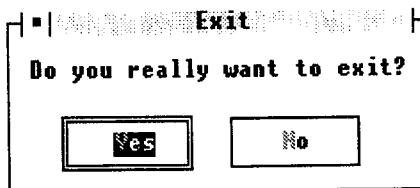


Figure 3.49 Exit Confirmation Form

## 4 COMPUTATIONAL METHODOLOGY

The SECPOP90 code uses the 1990 block-level census data to calculate the population counts for each of the user-defined MACCS grid (rosette) sections, 1990 county-level census data to calculate land fraction data, and 1990 county-level census data and 1992 county-level agricultural census data to calculate economic data. The methodology involved can be broken down into (1) the algorithms used within the code to accomplish this purpose and (2) the database structures used for the block and county-level databases. Sections 4.1 through 4.3 discuss the algorithms in detail. Section 4.4 discusses the database structures.

### 4.1 Population Count and Density Algorithms

The SECPOP90 code uses the user-defined MACCS grid for a user-defined site to establish a set of longitude-latitude boundaries within which the grid lies. The census data file is a binary file sorted primarily by descending longitude (west to east) and secondarily by descending latitude (north to south). An iterative algorithm is used (within the POINTR subroutine) to find the westernmost point in the census data file which lies on or to the east of the western longitude boundary. For that data point and each subsequent data point read from the census data file, it is determined if the point lies between the north and south latitudinal boundaries. When a point is found to lie between the established boundaries, the distance of that point from the site is calculated using the distance formula to determine if in fact the point lies within the outer limit of the MACCS grid. If the point meets the distance criteria, it is then processed to determine the exact grid element in which it lies based on its radial distance and direction from the site. The population associated with that data point is then added to the population in the appropriate element of the array TOTPOP. The array TOTPOP contains an array element for each grid section.

The algorithm used to determine the longitude-latitude boundaries of the grid will be discussed in section 4.1.1. The iterative algorithm used to determine the westernmost point in the census data file lying on or to the east of the western longitude boundary is discussed in section 4.1.2. Finally, the algorithms used to determine if a data point lies within the MACCS grid and to determine the specific grid element within which any point lies will be discussed in section 4.1.3.

Regional calculations use all of the same algorithms described below except that there is no need to determine where in the circle being calculated the population lies. The population density is calculated directly using the population and area of each census block.

#### 4.1.1 Boundaries Algorithm

The heart of the algorithm lies within the subroutine GETDIS which finds the distance (km) per degree latitude and distance per degree longitude for a specific geodetic latitude. Within the GETDIS subroutine, the geodetic latitude is first converted to radians. The corresponding latitude in the master coordinate system, geocentric, is then calculated using the following equation

$$\theta_{GC} = \tan^{-1} \left[ \left( \frac{R_{ep}}{R_{eq}} \right)^2 \tan \theta_{GD} \right]$$

where

$\theta_{GC}$	= the geocentric latitude,
$R_{ep}$	= the earth polar radius (km),
$R_{eq}$	= the earth equatorial radius (km), and
$\theta_{GD}$	= the geodetic latitude

The geocentric radius,  $R_{GC}$ , is then calculated using the equation

$$R_{GC} = \frac{R_{eq} R_{ep}}{\sqrt{R_{eq}^2 \sin^2 \theta_{GC} + R_{ep}^2 \cos^2 \theta_{GC}}}$$

The distance (km) per degree latitude, DPDLAT, and the distance (km) per degree longitude, DPDLON are derived using the following equations

$$DPDLAT = R_{GC} \frac{\sin \theta_{GC}}{\sin \theta_{GD}} \cdot \left( \frac{\pi}{180} \right)$$

$$DPDLON = R_{GC} \cos \theta_{GC} \cdot \left( \frac{\pi}{180} \right)$$

When finding the west and east longitude boundaries of the grid, the following assumptions are made and can be shown to be valid. If the west-east boundaries are established on a "horizontal" circumference of the earth through the site, the entire circular MACCS grid will lie between those same two longitudinal boundaries.

The distance to those longitude boundaries can be found by dividing the greatest radial distance on the MACCS grid by DPDLON, the distance per degree longitude at the site geodetic latitude.

Finding the north and south latitude boundaries is somewhat more complex. The distance per degree latitude decreases with increasing latitude. To make sure that the boundaries set will encompass the entire grid, distance per degree latitude, DPDLAT, is determined at the Tropic of Cancer for the half of the grid lying below the latitude of the site, and the distance per degree latitude, DPDLAT, determined at the site is used for the half of the grid north of the site latitude. As a result of this conservative approach, the north and south boundaries are well outside the grid, but this is acceptable since the boundaries are used only to determine which of the census data points will undergo further processing to determine if in fact they do lie within the grid.

#### 4.1.2 First Element Location Algorithm

An iterative "divide and conquer" algorithm is used in the subroutine POINTR to determine the first element in the census data file which lies on or to the east of the western longitude boundary of the MACCS grid. The set of records in the census data file is divided into two parts, and it is determined in which part the western boundary lies with the procedure then being repeated concentrating on that part of the set. This process is continued until one of two situations occurs. If two adjacent points are found which straddle the boundary, the easternmost record is marked. If a point is found which lies on the western boundary, the records are searched backward sequentially until the first of the data elements is found which lies on that boundary and that point is marked.

#### 4.1.3 Specific Grid Element Determination

When a census data element is found to lie within the longitude and latitude boundaries of the grid, both the distance per degree latitude, DPDLAT, and distance per degree longitude, DPDLON, are determined for the geodetic latitude of that data element. An average distance per degree latitude is determined between the census data element and the site. This average is used to calculate an actual distance in kilometers separating the latitudes of the element and the site. In turn, the distance in kilometers between the longitude of the element and the longitude of the site at the latitude of the element is calculated. The distance formula is then applied to determine the straight line distance between the site and the element. If the distance is less than the maximum user-defined radial distance, then each of the radial elements is examined to determine in which radial element the data element lies.

Once it is determined that a data element lies within the MACCS grid, it is then necessary to determine in which of the 16 directional grid elements the data element lies. The angle a line from the site to the data element makes with north is found as follows:

$$\theta_{de} = \tan^{-1} \left( \frac{x}{y} \right)$$

where

$\theta_{de}$	= the angle made between a line from the site to the data element and true north,
$x$	= the distance from the longitude of the data element to longitude of the site, and
$y$	= the distance from the latitude of the data element to the latitude of the site

This value,  $\theta_{de}$ , is then used to determine the specific directional element in which the data element lies.

## 4.2 Land Fraction Algorithms

In addition to location and population, every record in the block-level database also includes the area of the block and a code to indicate which county in the U.S. the block resides. This additional information is used by both the land fraction algorithm and the economic factors algorithms to estimate land fractions and economic factors respectively.

A county-level database also exists that contains the land fraction data for every county in the continental U.S. These data were obtained from the same 1990 census data files as the block-level data (PL 94-171).

The area of the blocks cannot be used to determine section land fractions directly for two reasons. First, the area given is only the land mass — no area for the water mass is given at the block level. Second, there is no simple way to aggregate the block areas to determine how much of a rosette section they “fill up” since the geometry of the blocks is unknown. Instead the area of the blocks is used to weight the county-level land fraction data.

During a site-specific calculation, a running sum for each rosette section is made of the total area of the blocks that lie within each section and a running sum is made of all of the weighted land fraction data. At the end of the calculation, the sum of the weighted land fractions is divided by the sum of the block areas. This is equivalent to the following formula for the land fraction for rosette section  $i, j$ :

$$FRCLND(i, j) = \frac{\sum_n A_{Block} \times FRCLND(County_{Block})}{\sum_n A_{Block}}$$

where

$FRCLND(i, j)$	= the estimated land fraction for the rosette section defined by the $i$ th direction and $j$ th and $j - 1$ radii (unless $j = 1$ , then the section is defined only by $i$ and $j$ — i.e., the innermost sections of the rosette),
$FRCLND(County_{Block})$	= the land fraction of the county that the present census block resides in,
$A_{Block}$	= the area of the present census block, and
$n$	= the number of census blocks that reside in section $i, j$ of the rosette

Note in areas where census blocks tend to be large — lakes, desert, national and state parks — it is possible that a census block centroid will not lie within a rosette section and the resulting land fraction will be estimated to be 0. While this is appropriate in a lake, it does not represent desert regions accurately. The user may need to edit the output data manually in such cases to obtain a better estimate.

## 4.3 Economic Factors Algorithms

SECPOP90 estimates the economic factors that are defined in MACCS Site Data File. The economic factors are calculated for the user-defined (or default) economic regions. The region inside the innermost radius is normally considered to be the exclusion region, while SECPOP90 calculates values for this region they are likely to be quite a bit different from the real values for the exclusion area if that area contains most of the nuclear power reactor site.

The algorithm used to calculate the economic factors is very similar to that used in section 4.2. The only difference is that the values are accumulated for each economic region instead of each rosette section. Likewise the county-level database has values for each economic factor for each county. Each Economic Factor (EF) for each economic region  $i$  is calculated using the following algorithm:

$$EF(i) = \frac{\sum_n A_{Block} \times EF(County_{Block})}{\sum_n A_{Block}}$$

where

$EF(i)$	= the estimated economic factor for the $i$ th economic region,
$EF(County_{Block})$	= the economic factor of the county that the present census block resides in,
$A_{Block}$	= the area of the present census block, and
$n$	= the number of census blocks that reside in economic region $i$

The economic factors estimated and the sources of the data are as follows:

FRMFRC	Fraction of land devoted to farming in the region, 1992 Census of Agriculture, Geographic Area Series 1B, County-Level Data, U.S. Department of Commerce, Economics and Statistics Administration, Bureau of the Census
DPF	Fraction of farm sales resulting from dairy production in the region, 1992 Census of Agriculture, Geographic Area Series 1B, County-Level Data, U.S. Department of Commerce, Economics and Statistics Administration, Bureau of the Census
ASFP	Annual average farm sales for the region (\$/hectare); 1992 Census of Agriculture, Geographic Area Series 1B, County-Level Data, U.S. Department of Commerce, Economics and Statistics Administration, Bureau of the Census
VFRM	Average farmland value for the region (\$/hectare); 1992 Census of Agriculture, Geographic Area Series 1B, County-Level Data, U.S. Department of Commerce, Economics and Statistics Administration, Bureau of the Census
VNFRM	Average nonfarm value for the region (\$/person), MACCS methodology using 1990 information from the 1993 and 1994 Statistical Abstract of the United States, U.S. Department of Commerce, Economics and Statistics Administration, Bureau of the Census, and 1990 county per capita income from 1994 County and City Data Book, U.S. Department of Commerce, Bureau of the Census, Data User Services Division

## 4.4 Database Structure

The two main databases used in SECPOP90 are the block-level census and county-level census databases, CENSUS90.DAT and COUNTY90.DAT. The block-level census database contains over six million records — a record for every census block in the continental U.S. The county-level census database contains over three thousand records, one for every county in the continental U.S. The sections below describe the databases in more detail.

### 4.4.1 Block-Level Database

The 1990 block-level census data is stored in a binary file, CENSUS90.DAT, which contains 6,660,337 records. Each record is 12 bytes long and contains the following 5 pieces of information:

- (1) a 2-byte integer code for the longitude of the geometric centroid of the census block,
- (2) a 2-byte integer code for the latitude of the geometric centroid of the census block,
- (3) a 2-byte integer for the residential population that resides within the census block,
- (4) a 4-byte integer for the land area ( $0.001 \text{ km}^2$ ) of the census block, and
- (5) a 2-byte integer code for the county that the census block resides in.

The integer codes for the longitude and latitude were derived to reduce the storage required for the block-level database. The longitude can be calculated using the following formula:

$$\text{longitude} = (\text{integer\_code} + 91993) / 1000.0$$

The latitude can be calculated using the following formula:

$$\text{latitude} = (\text{integer\_code} + 16610) / 1000.0$$

The integer code for the county is the index to the county-level data base and is the first field in each record of that database. More information on the county-level database can be found in the next section. Details on how the block-level database was constructed and verified can be found in section 5.1.

### 4.4.2 County-Level Database

The 1990 and 1992 county-level census data is stored in a fixed-width format ASCII text file, COUNTY90.DAT, which contains 3,111 records and one header line. Each record is one line long and contains the following 10 pieces of information:

CountyCode	the index to the county-level database
State	the two-letter abbreviation of the state in which the county lies,

CountyName	the name of the county,
FRCLND	the fraction of area that is land in the county,
FRMFRC	fraction of land devoted to farming in the county,
DPF	fraction of farm sales resulting from dairy production in the county,
ASFP	annual average farm sales for the county (\$/hectare)
VFRM	average farmland value for the county (\$/hectare)
VNFRM	average nonfarm value for the county (\$/person)
Notes	various notes concerning exceptions to the data. See Appendix C for a full listing of the database and an explanation of all notes

**Table 4.1 Excerpt From the County-level Database**

CountyCode	State	CountyName	FRCLND	FRMFRC	DPF	ASFP	VFRM	VNFRM	Notes
1	ME	Androscoggin County	0 945790	0 206791	0.125332	3216	4532	109414	0
2	ME	Aroostook County	0 976975	0 078229	0.027361	908	2062	93633	0
3	ME	Cumberland County	0 686591	0 100777	0.233237	730	6779	144689	0
:	:	:	:	:	:	:	:	:	:
3109	CA	Ventura County	0 835891	0 271368	0 001479	5147	17519	145082	1
3110	CA	Yolo County	0 989805	0 800826	0 002811	1115	5551	133615	1
3111	CA	Yuba County	0 979641	0 581876	0 029790	1142	6226	84089	0



## 5 SOFTWARE DEVELOPMENT

### 5.1 Block-Level Census Database

The block-level data and some county-level data (area of land and water) were extracted from the CD-ROM set of the *Census of Population and Housing, 1990 Public Law (PL) 94-171, Data Technical Documentation / prepared by the Bureau of the Census --Washington The Bureau, 1991*[sic]. This section describes how the data for the block-level census database were extracted from these CD-ROMs and how after each step of the process the results were verified and validated to ensure the integrity of the intermediate and final data.

#### 5.1.1 Construction of the Block-Level Census Database

The PL 94-171 files are stored on ten CD-ROMs in dBase III format (dBase). The first step in the construction of the census-level database (CENSUS90 DAT, located in the secpop90\census directory) was to extract the necessary information from the dBase files and store it in an unencoded binary format. There is a dBase file for each state and the District of Columbia. The name of these files is PL9417nn.DBF, where *nn* is the postal code for the state or district. The format of the dBase census records is specified in another dBase file, PL94STRU DBF, which is also included on the CD-ROMs. Table 5.1 shows the format of the state files. Note that all fields are of type character string.

Table 5.1 dBase III Census Record Format

Name	Type	Length	Decimal Count	Description
FILEID	C	8	0	File Identification
STUSAB*	C	2	0	State/US Abbreviation
SUMLEV*	C	3	0	Summary Level
GEOCOMP	C	2	0	Geographic Component
CHARITER	C	3	0	Characteristic Iteration
LOGRECNU	C	6	0	Logical Record Number
LOGRECPN	C	4	0	Logical Record Part Number
PARTREC	C	4	0	Total Number of Parts in Record
ANRC	C	2	0	Alaska Native Regional Corporation
AIANACE	C	4	0	American Indian/Alaska Native Area (Census)
AIANAPP	C	5	0	American Indian/Alaska Native Area (FIPS)
AIANACC	C	2	0	American Indian/Alaska Native Area Class Code
ARTLI	C	1	0	American Indian Reservation Trust Land Indicator Code
BLCK	C	4	0	Block
BLCKGR	C	1	0	Block Group
TRACTBNA	C	6	0	Census Tract/Block Numbering Area
CONGDIS	C	2	0	Congressional District (101st Congress)
CONCITCE	C	1	0	Consolidated City (Census)
CONCITFP	C	5	0	Consolidated City (FIPS)
CONCITCC	C	2	0	Consolidated City Class Code
CONCITSC	C	2	0	Consolidated City Population Size Code
CMSA	C	2	0	Consolidated Metropolitan Statistical Area
CNTY*	C	3	0	County
CNTYSC	C	2	0	County Population Size Code
COUSUBCE	C	3	0	County Subdivision (Census)
COUSUBFP	C	5	0	County Subdivision (FIPS)
COUSUBCC	C	2	0	County Subdivision Class Code
COUSUBSC	C	2	0	County Subdivision Population Size Code
DIVIS	C	1	0	Division
EXTCITIN	C	1	0	Extended City Indicator
INTUC	C	15	0	Internal Use Code
MSACMSA	C	4	0	Metropolitan Statistical Area/Consol Metro Statist Area
MSACMSAS	C	2	0	MSA/CMSA Population Size Code
PLACECE	C	4	0	Place (Census)
PLACEFP	C	5	0	Place (FIPS)
PLACECC	C	2	0	Place Class Code

Table 5.1 (Continued) dBase III Census Record Format

Name	Type	Length	Decimal Count	Description
PLACEDC	C		1	0 Place Description Code
PLACESC	C		2	0 Place Population Size Code
PMSA	C		4	0 Primary Metropolitan Statistical Area
REG	C		1	0 Region
STATECE*	C		2	0 State (Census)
STATEFP	C		2	0 State (FIPS)
URBANRUR	C		1	0 Urban/Rural
URBAREA	C		4	0 Urbanized Area
UASC	C		2	0 Urbanized Area Population Size Code
SAC1	C		5	0 Special Area Code (1)
SAC2	C		5	0 Special Area Code (2)
SAC3	C		4	0 Special Area Code (3)-Voting District Code
SAC4	C		4	0 Special Area Code (4)
SAC5	C		3	0 Special Area Code (5)
SAC6	C		3	0 Special Area Code (6)
SAC7	C		2	0 Special Area Code (7)
SAC8	C		2	0 Special Area Code (8)
SAC9	C		1	0 Special Area Code (9)-Land/Water Area Code
SAC10	C		1	0 Special Area Code (10)-Actual/Pseudo Voting District Code
AREALAND*	N	10		0 Area (land)
AREAWAT*	N	10		0 Area (water)
ANPSADPI*	C	66		0 Area Name/PSAD Term/Part Indicator
FUNCSTAT	C		1	0 Functional Status Code
GCUNI	C		1	0 Geographic Change User Note Indicator
HU100	N	9		0 Housing Unit Count (100%)
INTPTLAT*	C		9	0 Internal Point (latitude)
INTPTLNG*	C		10	0 Internal Point (longitude)
PARTFLAG	C		1	0 Part Flag
PSADC	C		2	0 Political/Statistical Area Description Code
POP100*	N	9		0 Population Count (100%)
SPFLAG	C		1	0 Special Flag
P001_0001	N	9		0 Population Table 1, Item 1
P002_0001	N	9		0 Population Table 2, Item 1
P002_0002	N	9		0 Population Table 2, Item 2
P002_0003	N	9		0 Population Table 2, Item 3
P002_0004	N	9		0 Population Table 2, Item 4
P002_0005	N	9		0 Population Table 2, Item 5
P003_0001	N	9		0 Population Table 3, Item 1
P003_0002	N	9		0 Population Table 3, Item 2
P003_0003	N	9		0 Population Table 3, Item 3
P003_0004	N	9		0 Population Table 3, Item 4
P003_0005	N	9		0 Population Table 3, Item 5
P004_0001	N	9		0 Population Table 4, Item 1
P004_0002	N	9		0 Population Table 4, Item 2
P004_0003	N	9		0 Population Table 4, Item 3
P004_0004	N	9		0 Population Table 4, Item 4
P004_0005	N	9		0 Population Table 4, Item 5
P004_0006	N	9		0 Population Table 4, Item 6
P005_0001	N	9		0 Population Table 5, Item 1
P005_0002	N	9		0 Population Table 5, Item 2
P005_0003	N	9		0 Population Table 5, Item 3
P005_0004	N	9		0 Population Table 5, Item 4
P005_0005	N	9		0 Population Table 5, Item 5
P005_0006	N	9		0 Population Table 5, Item 6
H001_0001	N	9		0 Housing Table 1, Item 1

\* Indicates field used in block or county-level census database

To extract the necessary information from the dBase state files, a highly modified version of the Microsoft QuickBasic program included in the CD-ROMs' README file (see Figure 5.1) was written, CONVERT1.BAS (see Appendix G). This Basic program extracted both the block and county-level information and stored it in two separate files, BLOCK\_1.BIN and COUNTY\_1.TXT respectively. The data stored in BLOCK\_1.BIN was converted from the dBase text format to Visual Basic's native binary representation. Each record in BLOCK\_1.BIN was 19 bytes long and contained the longitude and latitude of the census block geographical centroid (4 bytes each), population (4 bytes), area (4 bytes), state census number (1 byte), and county census number (2 bytes).

**Figure 5.1 PL94-71 README File dBase to ASCII Conversion Program**

The second part of the block-level conversion process was to create a second binary file, BLOCK\_2.BIN that contained the same information as BLOCK\_1 BIN but in an encoded format so that the data records would be smaller (BLOCK\_1 BIN was 126 Mbytes, BLOCK\_2 BIN was 80 Mbytes) and easier to use (see Appendix G for the CONVERT2.BAS Visual Basic code). The encoding for the latitude and longitude is discussed in section 4.4.1. The encoding of the county code was performed using a lookup table in the CONVERT2 BAS program. This was necessary because the county code used in the census uses designators that are unique within each state but not throughout the country. The code generated for the county designator of each block is also the county index to the county-level database. The codes were ordered by the census state-level designator as the primary key, and the census county designator as the secondary key. Furthermore, since each county had a unique designator in BLOCK\_2 BIN, the state designator was not stored in BLOCK\_2 BIN. That information can be obtained by performing a table lookup in the county-level database. SECPOP90 has no need for that information at present.

The next part of the creation of the block-level census database was to sort the records by longitude and latitude. This was done in a two-step process using two FORTRAN programs, SORTBIN1.FOR and SORTBIN2.FOR (see Appendix G). FORTRAN was used because it is faster and can support more open files. SORTBIN1.FOR first sorted the BLOCK\_2.BIN file by separating it into 30 separate files based on a census block's longitude. SORTBIN2.FOR then individually sorted the 30 files by longitude and latitude, writing the results of each sorted file to BLOCK\_3.BIN. The SORTBIN2.FOR program contains an error. It incorrectly sorts files that contain both records with positive and negative longitudes. (Negative and positive longitudes were present in the intermediate BLOCK\_n.BIN files because of the encoding of the longitude from 32 to 16 bit integers.) Fortunately, this case only existed in one of the 30, BLOCK\_2.BIN files. The database was corrected during the Verification and Validation (V&V) testing (see section 5.1.2). The SORTBIN2.FOR program was not corrected and still contains the error.

The final step was the concatenation of duplicate records by the FORTRAN program RMOVEDUP.FOR (see Appendix G). Two records were considered duplicates if they shared the same longitude, latitude, and county code. When such records were found, their areas and population counts were added together and a single record was created to represent the duplicate records. The final resulting database was written to BLOCK\_4 BIN, which was then renamed to CENSUS90.DAT.

### **5.1.2 Verification and Validation of the Block-Level Census Database**

Upon completion of each of the steps described above, diagnostic output was produced to verify that the conversion programs were operating correctly, and additional testing was performed to validate that the block-level census data integrity was maintained. (No effort was made to determine the integrity of the initial data on the PL 94-171 CD ROMs. For a discussion of the Census Bureau's level of data confidence see Appendix E.)

After the first conversion of the dBase CD-ROM data, the TEST1 BAS Basic program (see Appendix H) was executed to read in all of the census block records and print out a summary of all of the county and state total populations and land areas. The summary results agreed with the state and county populations and were always less than or equal to the state and county land areas. (Aggregated block-level census land areas are usually less than county or state land areas because of the differences in tabulating the two area types. See Appendix E.) Comparisons were done using a Microsoft Excel spreadsheet. A sample of Rhode Island census block longitudes and latitudes from the CD-ROM database files was compared with those in BLOCK\_1 BIN and were in agreement.

After the second conversion of the BLOCK\_1.BIN data and the creation of the BLOCK\_2 BIN database, the TEST2.BAS program (see Appendix H) was executed to print out a sample of 71 records from both the BLOCK\_1 BIN and BLOCK\_2 BIN databases. The results were examined by hand and were found to be in agreement. The file lengths of the two databases were also examined to ensure that they contained the same number of records.

After the sorting of the BLOCK\_2.BIN database was completed and the BLOCK\_3.BIN file was generated, the TEST3 BAS program (see Appendix H) was executed to print out the records for Washington County, Maine. These were compared with the same records found in the CD-ROM database files and were in agreement. The file lengths of the two databases, BLOCK\_2 BIN and BLOCK\_3 BIN, were compared to ensure that they contained the same number of records. The file lengths were equal.

A repeat of testing done after the first conversion was performed on the final BLOCK\_4.BIN database using a modified version TEST1.BAS named TEST4 BAS (see Appendix H). The summary results agreed with the original data files from the PL 94-171 CD-ROMs. The Rhode Island data also agreed and the number of records remaining in BLOCK\_4.BIN was equal to the initial number of records minus the duplicate records. BLOCK\_4.BIN's name was changed to CENSUS90 DAT, which did not alter the data in any way. A final test to ensure that the database records were sorted correctly, TEST5.BAS (see Appendix H), was performed. It was discovered that a section of records were out of order. Two programs, FIX1 BAS and FIX2 BAS (see Appendix H), were run to further diagnose and correct the database. TEST5 BAS was rerun. The error was corrected and there were no other records found out of order.

## **5.2 County-Level Census Database**

The county-level census database was constructed with data from several sources. The PL 94-171 CD-ROM data files (see 5.1.1), *Census of Agriculture, 1992: Final County Files [machine-readable data file]* / conducted by the Bureau of the Census Washington: The Bureau [producer and distributor], 1993 [sic], *County and City Data Book 1994 on CD-ROM [machine-readable data files]* / prepared by the Bureau of the Census--Washington: The Bureau [producer and distributor], 1995 [sic], and 1993, 1994 *Statistical Abstract of the United States*, U.S. Department of Commerce, Economics and Statistics Administration, Bureau of the Census.

### **5.2.1 Construction of the County-Level Census Database**

Land and water areas were extracted from the PL 94-171 CD-ROM data files (see section 5.1.1 for how this was accomplished). The area data were then imported into a Microsoft Excel spreadsheet and the county land fraction, FRCLND, was then calculated by dividing the land area by the sum of the land and water area. The county population data were also extracted for verification purposes but it was not used in the county-level database. There were no exceptions, omissions, or special treatment for any of the data items extracted from the PL 94-171 CD-ROM data files.

The agricultural data were extracted from the 1992 Census of Agriculture CD-ROM Geographic Area Series 1B using a program that was included on the CD-ROM called EXTRACT. The EXTRACT program allowed the data items described in Table 5.2 to be extracted for each state and saved to a file that was then imported into Microsoft Excel.

It is important to note that for many of the counties one or more of the above data items was not available due to the confidentiality requirements of the census (see Appendix E). In most cases estimates were made for the undisclosed amount by distributing the excess acreage or dollars evenly among the undisclosed farms. The excess acreage or dollars were determined by subtracting the sum of the disclosed amounts from the state totals provided. This method was chosen because the number of farms and dairy farms in a county was always provided, allowing the estimates to be made in a straightforward manner. Other more complicated methods such as apportioning the excess by acreage or estimating county values by surrounding county values did not provide any more reasonable or trusted estimates. In a small number of cases (particularly when undisclosed values were for small farms in heavily populated areas, e.g., Queens, New York, or San Francisco, California), it was clear that average dollars per farm or acre were excessive. Those exceptions were handled on a case-by-case basis and are described in the notes at the end of Appendix C. The District of Columbia has no agricultural products.

**Table 5.2 Census of Agriculture Data Items**

Item	Description
010001	Farms (number)
010002	Land in farms (acres)
010004	Est mkt val land & bldg@1 avg/farm (\$)
010006	Est mkt val all mach@1 avg/farm (\$)
010019	Mkt val of ag products sold (\$1,000)
020064	Dairy products, (farms)
020065	Dairy products, (\$1,000)

From the Census of Agriculture Data, four agricultural values (these are the agricultural economic values defined for MACCS, see Appendix B) were calculated for each county. They are as follows

- FRMFRC Fraction of land devoted to farming in the county,
- DPF Fraction of farm sales resulting from dairy production in the county,
- ASFP Annual average farm sales for the county (\$ / hectare), and
- VFRM Average farmland value for the county (\$ / hectare). This includes the average value of farm land, buildings, and machinery

The fifth economic value calculated for each county is the average of nonfarm value (\$ / person), VNFRM (also a MACCS economic value, see Appendix B). The methodology used to calculate this value for each county is equivalent to that described by Sprung et al.<sup>1</sup> First a value of VNFRM is calculated for the entire United States. Then estimates of county values are made using the following equation

$$VNFRM_{County} = VNFRM_{US} \left( \frac{PCI_{County}}{PCI_{US}} \right)$$

where

- $VNFRM_{County}$  = the average non-farm value for the county (\$ / person),
- $VNFRM_{US}$  = the average non-farm value for the United States (\$ / person),
- $PCI_{County}$  = the per capita income for the county (\$ / person), and
- $PCI_{US}$  = the per capita income for the United States (\$ / person)

The values for the per capita income were extracted from the 1994 County and City Data Book CD-ROM database files. They were imported directly into Microsoft Access and then modified to contain only the per capita information and the associated footnotes that indicated exceptions and omissions to the data. From Microsoft Access the data were then imported into a Microsoft Excel spreadsheet where they were incorporated into the rest of the county-level database. The exceptions to the data consisted of incidents where two or more counties or independent cities (countylike entities that have no other county affiliation) reported per capita income figures jointly. In all such cases, equal per capita income amounts were applied to all joined counties. For the exact counties affected, see Appendix C.

$VNFRM_{US}$  is calculated using the equation

$$VNFRM_{US} = \frac{RTW_{US} + VSL_{US} - VFA_{US} + VFHP_{US}}{POP_{US}}$$

where

$RTW_{US}$	= the reproducible tangible wealth for the United States (\$),
$VSL_{US}$	= the value of suburban land in the United States (\$),
$VFA_{US}$	= the value of farm assets in the United States (\$),
$VFHP_{US}$	= the value of farm household possessions in the United States (\$), and
$POP_{US}$	= the population of the United States (persons)

$VSL_{US}$  is calculated using the equation

$$VSL_{US} = UBL_{US} \cdot MHV_{US} \cdot LPA_{US} \cdot FLV_{US}$$

where

$UBL_{US}$	is the amount urban and built-up land in the United States (acres),
$MHV_{US}$	is the median housing value for the United States (\$ / house),
$LPA_{US}$	is average number of housing lots per acre (houses / acre), and
$FLV_{US}$	is the average fraction of house cost due to land value

The values and the sources for the variables used in producing  $VNFRM_{US}$  were all obtained from the 1993 and 1994 *Statistical Abstract of the United States* except for  $LPA_{US}$  and  $FLV_{US}$  which were obtained from NUREG/CR 4551 (see end note 1) and are equal to 5.0 houses / acre and 0.2 respectively. For details on the remaining values see Appendix D

### 5.2.2 Verification and Validation of the County-Level Census Database

Unlike the construction of the block-level database, the construction of the county-level database did not require the creation of any additional programs for conversion or sorting. The tools used were Microsoft Excel, Microsoft Access, and the EXTRACT program provided by the Bureau of Census. It is assumed that these programs functioned properly, however, after every operation and / or manipulation of the county-level data, checks were made to ensure that each county had the proper number of records and that all individual county amounts totaled (or other appropriate operation) to the state aggregate amounts

## 5.3 Verification and Validation of the SECPOP90 Code

The verification and validation of the SECPOP90 code concentrated on the accuracy of the results of the site-specific population calculations. There are several justifications for this approach. First, the regional calculations use the same algorithms as the site-specific calculations, therefore the regional calculations are indirectly verified and validated by the verification and validation of the site-specific calculations. Second, there is no real way to validate the site-specific economic data that were derived from the county-level database. The economic values are estimates and should only be used for relative evaluations. There are no absolute measures for validating the economic data.

This section pertains to the verification and validation of the SECPOP90 code only. For a discussion of the code's development, see section 4

### 5.3.1 Internal Verification and Validation

The first step in the verification of the SECPOP90 code was to use the first 36 data points on the census data file to enable the code developer to readily verify that the population for each of those data points was correctly placed on a magnified grid close to the "site." The "site" was placed at each of the "corners" of the data as well as at the "middle" of the data. SECPOP90 was found to work correctly in each case.

The SECPOP90 code was then exercised using the 1990 census data and the results contrasted with the results when the SECPOP code was exercised using the 1980 census data for five sites. The sites selected were distributed throughout the continental United States. It was found that the differences were, for the most part, well within differences that would have been predicted simply because of the population growth over the last 10 years.

Final internal verification efforts involved comparing the results obtained from the VAX/VMS system FORTRAN code version with the results obtained using the PC Visual Basic version. The results were in good agreement.

### 5.3.2 External Verification and Validation

A survey of U S commercial nuclear power plants identified those plants that have provided the NRC with updated population information based on the 1990 census data. Licensees for 12 plants have provided this information. Of these 12 plants, four only provided the total population within a specific distance from the plant. The others provided the information in tables, graphs, or both, by rings of 16 segments. The ring distances vary among plants. Licensee submittals may not be accurate, a deviation from the licensee's number(s) is not necessarily an error. In several plants, the licensee's population totals within a ring or sector (or both in one case) have been found to be incorrect. One licensee appears to have averaged the number of people within a ring, which is not accurate.

The following table shows the results for the total populations as reported by the licensees, as calculated by SECPOP90, and as percent of reported  $[(\text{as calculated by SECPOP90}) / (\text{as reported by the licensees})] \times 100$

**Table 5.3 Total Reported and Calculated Populations**

Plant Name	Reported	SECPOP90	Percent of Reported
FitzPatrick <sup>2</sup>	44,231	39,443	89%
Millstone 1 <sup>3</sup>	3,363,745	2,780,995	83%
Monticello <sup>4</sup>	2,273,213	2,416,872	106%
*Nine Mile Point <sup>5</sup>	924,000	908,835	98%
*Oyster Creek <sup>6</sup>	96,718	101,433	105%
*Peach Bottom APS <sup>7</sup>	10,257,315	6,803,644	66%
DC Cook <sup>8</sup>	52,953	51,070	96%
Diablo Canyon <sup>9</sup>	374,701	376,012	100%
Fort Calhoun <sup>10</sup>	760,431	772,086	102%
Prairie Island <sup>11</sup>	25,342	24,969	99%
Turkey Point <sup>12</sup>	2,613,535	2,602,834	100%

\* These plants only provided total population within a radius

As shown above, on a total population basis, all of the results are in reasonable agreement. The program also provides the typical rosettes which show the populations in relation to the plant site. Examples of the rosettes for Millstone 1, Monticello, and Diablo Canyon are shown below. The results can also be provided as a MACCS site file for direct use in MACCS calculations. An example of this type of output is shown in section 3.6.3.

Also included are population tables (see Appendix F) generated from licensee information and SECPOP90 output. These tables show the actual population by ring segment and the percent difference from the licensee's reported values. As with all percent results, caution should be used when interpreting the information. For example, at DC Cook, ring 1, south sector, the licensee reported 5 people while SECPOP90 calculated 20 people, a 400% difference with no practical effects.

Generally, SECPOP90 calculates the total population in each ring and each sector reasonably well. However, the ring-segment population can have a large deviation from the licensee's values. Assuming that the licensee's values are accurate, this can be caused by the code's identification of the census block centroid. Wherever the census block centroid is located within the rosette, that is where all of the people for that census block will be counted. This can be seen in the Diablo Canyon tables in the NW and NNW segments, rings 10 through 50.

Another factor is the level of detail for the census data that the licensee used. If the level of detail was at the county, city, or block group level, the reported values will be different. As previously mentioned, one licensee (Monticello) used average data for rings 2 through 6. The reasonableness of the results in both the ring and sector attests to the fact that the ring-segment to ring-segment differences are not significant. The one exception seems to be low population sites with sporadic population, like Turkey Point. In these cases the percent differences become large. If these differences are correct, it is not clear what effect this may have on MACCS calculations. In such cases, engineering judgment is needed to assess the impact of the low population on the health effects of interest, with consideration being given to the meteorology selected (wind rose).

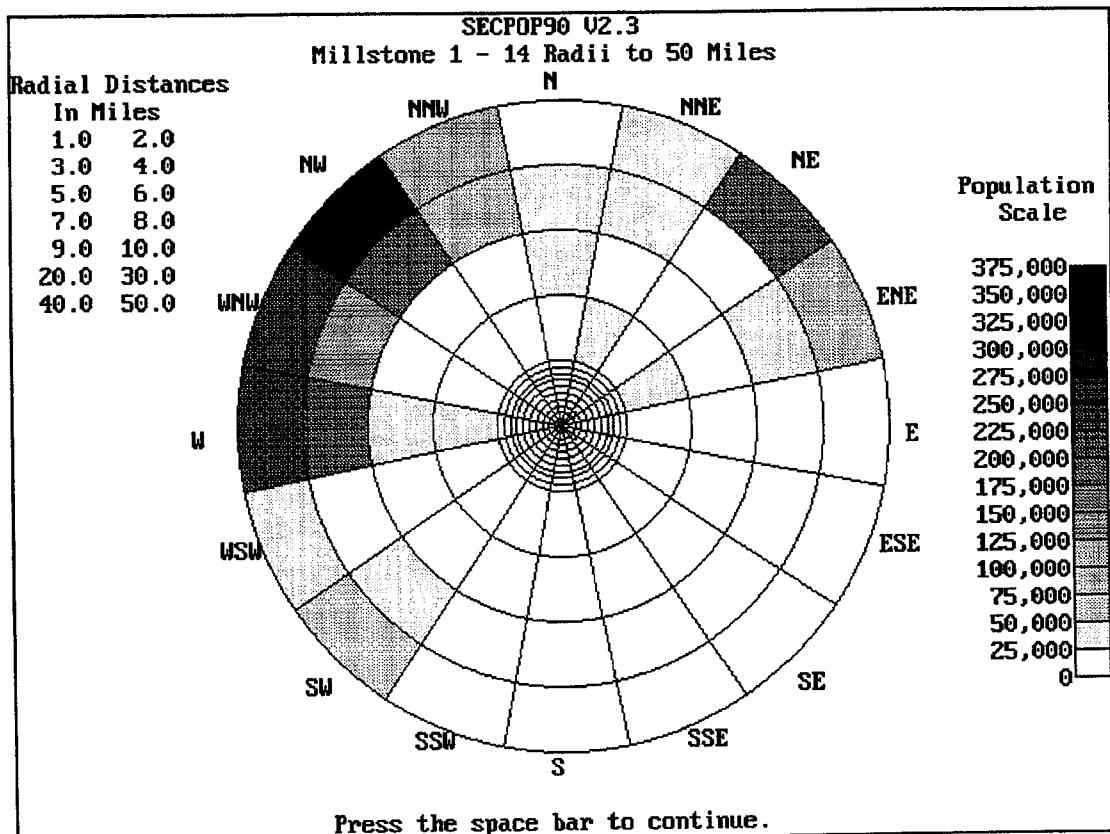


Figure 5.2 Millstone Rosette

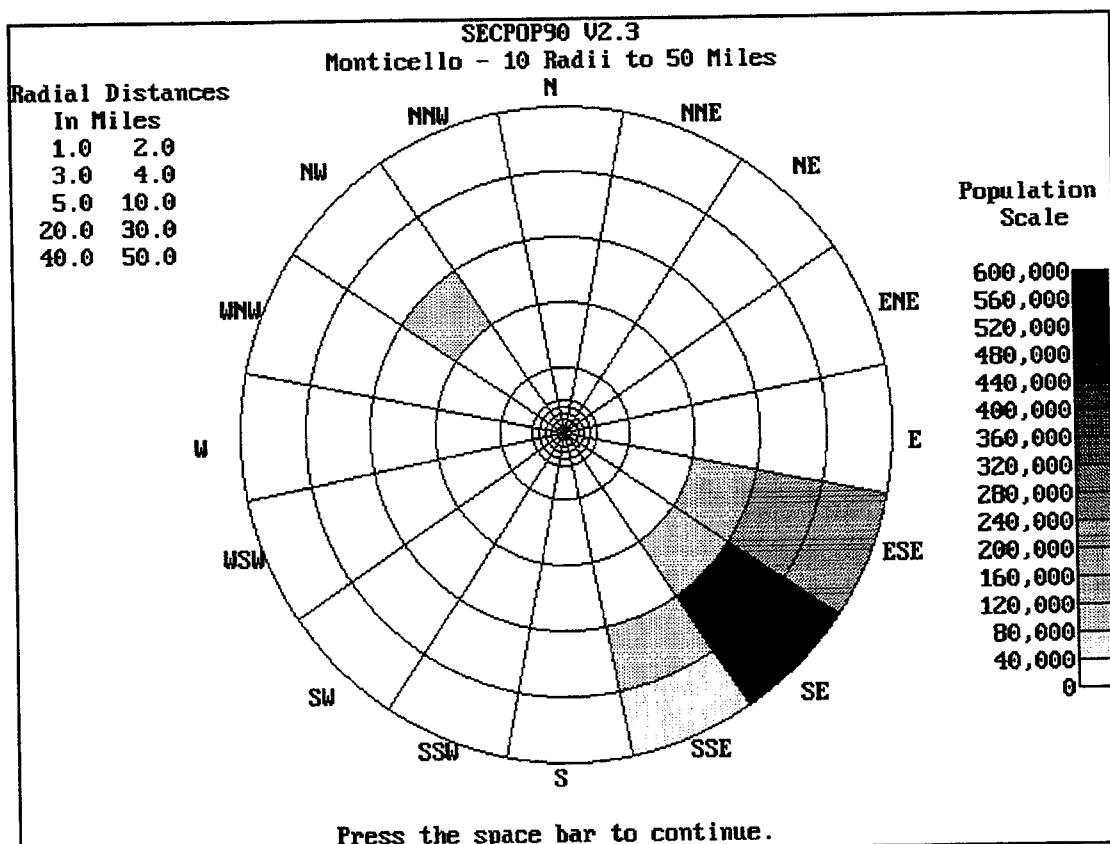


Figure 5.3 Monticello Rosette

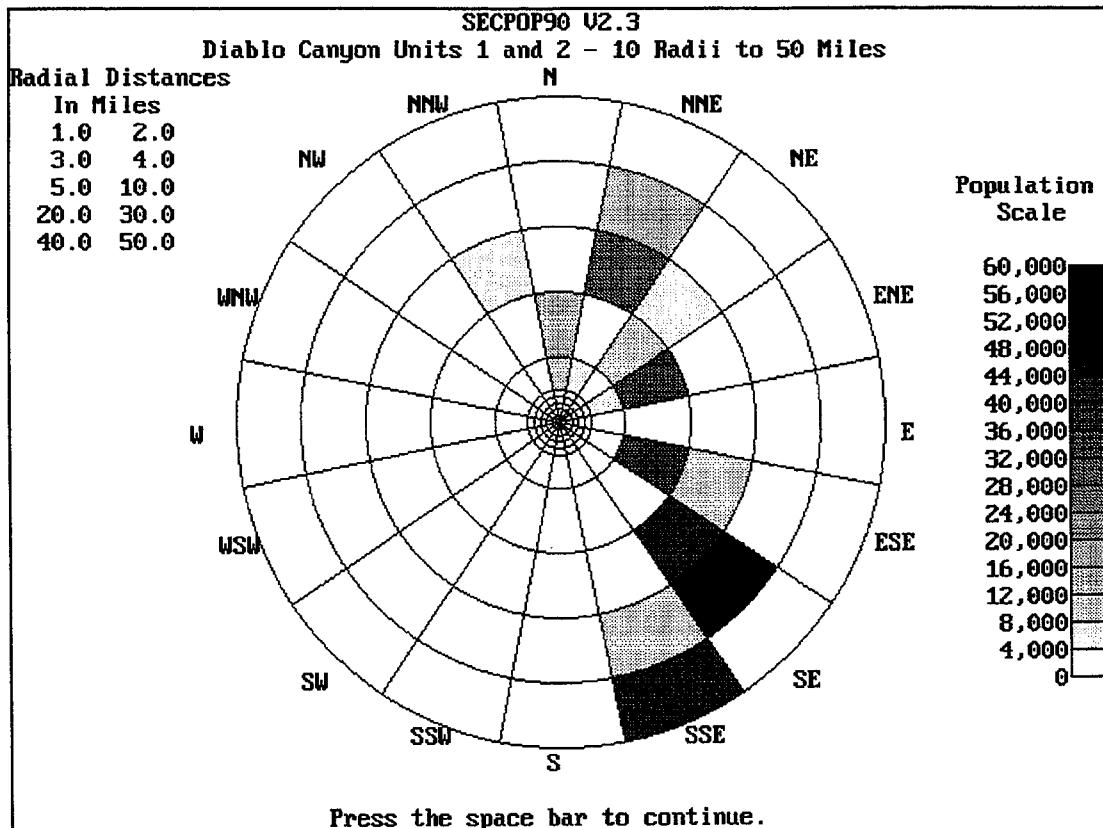


Figure 5.4 Diablo Canyon Rosette

---

<sup>1</sup> Sprung, J. L., Rollstin, J. A., Helton, J. C., Jow, H-N, Evaluation of Severe Accident Risks Quantification of Major Input Parameters, NUREG/CR — 4551, SAND86 — 1309, Vol2, Rev 1, Part 7, U.S Nuclear Regulatory Commission, Washington, DC, December 1990, pg 5-4

<sup>2</sup> James A FitzPatrick Emergency Plan, Volume 1, Revision 2, dated September 9, 1992

<sup>3</sup> Millstone 1 Updated Final Safety Analysis Report, Chapter 2, Revision 6, dated May 1993

<sup>4</sup> Monticello Final Safety Analysis Report, Tables 2 2-1 and 2 2-2, Revision 11, dated December 1991

<sup>5</sup> Nine Mile Point 2 Safety Evaluation Report, dated August 7, 1991, TAC Number 69095 [50 Mile radius]

<sup>6</sup> Oyster Creek, GPU submittal dated May 19, 1992, C321-92-2118 [10 Mile radius]

<sup>7</sup> Peach Bottom Atomic Power Station, Philadelphia Electric Company submittal dated May 21, 1992, Accession Number 9206020160 [60 Mile radius]

<sup>8</sup> DC Cook Final Safety Analysis Report, Page 2 1-8 and Figure 2 1-5, updated July 1993

<sup>9</sup> Diablo Canyon Final Safety Analysis Report, Chapter 2, Revision 8, dated September 1992

<sup>10</sup> Fort Calhoun Station Radiological Emergency Response Plan, Section J, Figures J-6 through J-8, Pages 24 through 26.

<sup>11</sup> Evacuation Time Estimates for the Prairie Island Nuclear Plant Plume Exposure Pathway Emergency Planning Zone, Table 3 1, Page 3-3, dated December 1992

<sup>12</sup> Turkey Point Final Safety Analysis Report, Tables 2 4-1 and 2 4-3, Revision 10, dated July 1992



## APPENDIX A — SITE LIST

This is the list of site files that are available in the secpop90\sites directory

ALLENS.SIT	ARKANSA.SIT	BAILLY.SIT	BLACKFOX.SIT	BRAIDWOO.SIT
BEAVER.SIT	BELLEFON.SIT	BIGROCK.SIT	BYRON.SIT	CALLAWA2.SIT
BROWNS.SIT	BRUNSWI2.SIT	BRUNSWIC.SIT	CHEROKEE.SIT	CLINTON.SIT
CALLAWAY.SIT	CALVERT.SIT	CATAWBA.SIT	DAVIS.SIT	DIABLO.SIT
COMANCHE.SIT	COOPER.SIT	CRYSTAL.SIT	FERMI.SIT	FITZPATR.SIT
DONALD.SIT	DRESDEN.SIT	DUANE.SIT	GINNA.SIT	GRAND.SIT
FORKED.SIT	FTCAL.SIT	FTSTVRA.SIT	HOPE.SIT	INDIAN.SIT
HADDEM.SIT	HARTSVIL.SIT	HATCH.SIT	LASALLE.SIT	LIMERICK.SIT
JOSPEH.SIT	KEWAANEE.SIT	LACROSSE.SIT	MIDLAND.SIT	MILLS.SIT
MAINE.SIT	MARTILE.SIT	MCGUIRE.SIT	NORTH.SIT	OCONEE.SIT
MILLSTON.SIT	MONTICEL.SIT	NINE.SIT	PALO.SIT	PEBBLE.SIT
OYSTER.SIT	PALISADE.SIT	PERRY.SIT	PHIPPS.SIT	POINT.SIT
PERKINS.SIT	QUAD.SIT	RANCHO.SIT	PILGRIM.SIT	ROBINSON.SIT
PRAIRIE.SIT	SANONO.SIT	SEABROOK.SIT	RIVER.SIT	SHEARON.SIT
SALEM.SIT	SKAGIT.SIT	SOUTHTEX.SIT	SEQUOYA.SIT	SURRY.SIT
SHOREHAM.SIT	THREE.SIT	TROJAN.SIT	STLUCIE.SIT	VERMONT.SIT
SUSQUEHA.SIT	VOGTLER.SIT	WATERFOR.SIT	TURKEY.SIT	WOLF.SIT
VIRGIL.SIT	WPPSS35.SIT	YANKEE.SIT	WATTS.SIT	ZIMMER.SIT
WPPSS.SIT			YELLOW.SIT	
ZION.SIT				



## APPENDIX B — MACCS SITE DATA FILE FORMAT

The following is taken directly from NUREG/CR 4691, *MELCOR Accident Consequence Code System (MACCS)*, Volume 1, pp. B-6 - B-13, February 1990 for the user's convenience. A listing of a sample Site Data File is supplied in this document as Table 3.1

### B.3 Site Data File Format

In the Site Data File, the user specifies the population distribution and land use information for the region surrounding the site. Contained in the Site Data File are the geometry data used for the site (spatial intervals), population distribution, fraction of the area that is land, watershed data for the liquid pathways model, information on agricultural land use and growing seasons, and regional economic information. An example of a Site Data File is provided in Appendix D.2.

The decision on whether to use a Site Data File is determined by a flag set in the EARLY User Input File (see Section 2.3). If a Site Data File is not being used, the EARLY User Input File determines the population density to be used by both the EARLY and CHRONC modules. It is not possible for the user to supply differing population data for the two modules.

The Site Data File used in MACCS is a formatted file. The data must appear exactly as described in this Appendix and in exactly the same order. In contrast to the User Input Files for MACCS, which are processed by a free format input processor, the Site Data File is processed with fixed format FORTRAN READ statements.

The use of fixed format READ statements requires that the user exercise special attention to line up the data items in their proper fields. Any numeric items specified in exponential format (e.g., 1.E-6) must be right-justified in the field.

In contrast to the User Input Files, where every value is validated by the code to ensure that it lies within a range of allowable values, the validation performed on the Site Data is only partially complete. Some of the input parameters on this file are rigorously checked to ensure that they fall within the allowed range, while other values are not checked at all. It is recommended that the user exercise scrupulous care in constructing a Site Data File. It is very important that all items appear in their proper fields and that all numeric values lie within the range of acceptable values. Failure to conform to these requirements may lead to the generation of spurious results.

The first two records of the Site Data File contain identification information. Up to 80 characters may be used on each line. This header information is printed on the output listing.

Following the descriptive text fields, there are six data cards which specify the amount of data which is being supplied on the file. The values defined on these data cards must be consistent with the MACCS model definition data defined by the ATMOS, EARLY, and CHRONC User Input Files. In other words, the value of the Site Data File input variable and the corresponding comparison variable must be identical. If any inconsistencies are detected, execution of the program will be terminated.

An example of the data cards described up to this point is included for the purpose of illustration.

MACCS SITE DATA FILE FOR SURRY (JLS, 11/10/88)  
 SECPOP POP DISTRIBUTION FROM 1980 CENSUS DATA ALTERED USING 0 10 MI NRC  
 26 SPATIAL INTERVALS  
 16 WIND DIRECTIONS  
 7 CROP CATEGORIES  
 4 WATER PATHWAY ISOTOPES  
 2 WATERSHEDS  
 59 ECONOMIC REGIONS

The format of the data is as follows:

<u>Line</u>	<u>Columns</u>	<u>Format</u>	<u>Input Variable</u>	<u>Identification</u>	<u>Comparison Variable</u>	<u>Allowed Range</u>
3	1-4	I4	NSPDTS	Number of spatial intervals	NUMRAD	2 - 35
4	1-4	I4	NWNNDIR	Number of wind direction sectors	NUMCOR	16 - 16
5	1-4	I4	NCPGZN	Number of crops to be considered	NFICRP	1 - 10
6	1-4	I4	NWPISO	Number of water pathway nuclides	NUMWPI	1 - 10
7	1-4	I4	NWTRSH	Number of watershed categories	NUMWPA	1 - 4
8	1-4	I4	NECRGN	Number of economic regions	N.A.	1 - 99

Eight blocks of site data follow the introductory block described above. Each of these data blocks is introduced by a separator line that identifies the block of data to follow. The first line of each data block must be the separator for that block. Since each separator is read with a format of 1X,A22, the first character of the separator line is ignored and the following 22 characters must match the identification field for that specific data block.

#### Geometry Data Block

Geometry information is used to specify the population and land use data. The geometry data consist of the distance in kilometers to the endpoints of the spatial intervals. This grid definition must agree with the grid used in the ATMOS User Input File (see Section 1.3). A relative error of 10 percent in the endpoint distances is allowed. For larger discrepancies in the geometry data, the error flag will be set and execution will terminate upon completion of the Site Data File input processing.

The first line of the geometry information contains the 22 character separator beginning with "SPATIAL DISTANCES" in column 2. Next, the endpoint distances in kilometers are specified, eight values per line, using the format described below. As many lines as are needed to define the spatial distances are used. The formatting of this information is illustrated below by presentation of the spatial distance data for a single wind direction sector.

Note: The spacing between adjacent spatial intervals should be at least 0.1 kilometer.

**SPATIAL DISTANCES**

0.16	0.52	1.21	1.61	2.13	3.22	4.02	4.83
5.63	8.05	11.27	16.09	20.92	25.75	32.19	40.23
48.28	64.37	80.47	112.65	160.93	241.14	321.87	563.27
804.67	1609.34						

The format used to process the data is as follows:

<u>Columns</u>	<u>Format</u>	<u>Input Variable</u>	<u>Information</u>	<u>Comparison Variable</u>	<u>Allowed Range</u>
1-80	8E10.2	SPDSTS	Spatial interval endpoints (km)	SPAEND	0.001 - 9999.0

Population Data Block

The population data for each element in the spatial grid is defined here. The first line of the data block contains the 22 character separator beginning with "POPULATION" in column 2. Next, the number of people in each element is given for the first sector (The first sector is assumed to be centered on North). The population data are specified, eight values per line, using the format described below. As many lines as needed to cover all the spatial elements in the sector are used. Proceeding in a clockwise rotation, the population data for the second ANNE) and subsequent sectors follow. Data for all 16 wind directions (sectors) must be provided. Data for each sector begins on a new line. The formatting of this information is illustrated below by presentation of the population data for a single wind direction sector for 26 radial distance elements.

**POPULATION**

0.	0.	0.	0.	0.	0.	4.	5.
6.	25.	3341.	7107.	2173.	0.	1305.	474.
2252.	2945.	5403.	20169.	112004.	3431358.	1355700.	2742710.
2487346.	104331.						

The format used to process the data is as follows:

<u>Columns</u>	<u>Format</u>	<u>Variable</u>	<u>Identification</u>	<u>Allowed Range</u>
1-80	8E10.0	POPDAT	Population data	0.0 - 1.E9

Land Fraction Block

The fraction of each spatial element that is land (as opposed to lakes, oceans, etc.) must be defined. The first line of the data block contains the 22 character separator beginning with "LAND FRACTION" in column 2. Next, the fraction of area that is land in each radial spatial interval of the first sector is given. All values must be between zero and one. A value of zero means the element has no land, a value of one means the element is all land. The land fraction data are specified, 16 values per line, with the format described below. As many lines as needed to define

all the spatial intervals in the sector are used. The land fraction data for the second and subsequent sectors follow in a clockwise rotation. Data for all 16 wind direction sectors must be provided. The data for each sector begins on a new line. The formatting of this information is illustrated below by presentation of the land fraction data for a single wind direction sector.

## LAND FRACTION

The format used to process the data is as follows:

<u>Columns</u>	<u>Format</u>	<u>Variable</u>	<u>Identification</u>	<u>Allowed Range</u>
1-80	16F5.2	FRCLND	Land fraction	0.0 - 1.0

### Region Identification Block

This data block identifies each of the spatial intervals with an economic region. A region can be a state, a county, or any division desired. A description of the economic data used for each region is given later in this Appendix. The first line in the datablock contains the 22 character separator beginning with "REGION INDEX" in column 2. The next line contains two digit integers associating a region index with each of the spatial elements in the first sector. The data are specified with the format defined below. All of the region indices for one sector will fit on one line. The region indices for the second and subsequent sectors are on the following lines, a new line for each sector in a clockwise rotation. A total of 16 lines are required. For example, a region index of 09 means that economic data for region number nine will be used for the spatial element. (See the section of this Appendix on Regional Economic Data.) The formatting of this information is illustrated below by a presentation of the region identification data for a single wind direction sector.

## REGION INDEX

The format used to process the data is as follows:

<u>Columns</u>	<u>Format</u>	<u>Variable</u>	<u>Identification</u>	<u>Allowed Range</u>
1-80	40I2	INDREG	Region index	1 - NECRGN

### Watershed Identification Block

Each of the spatial intervals in the grid must be associated with one of the watershed classes. The definition of the efficiency of these watershed classes is described later in this appendix in the "WATERSHED DEFINITION" block. The watershed identification data block begins with the 22 character separator beginning with "WATERSHED INDEX" in column 2. The next line contains two digit integers associating a watershed type with each of the spatial elements in the first sector. The data are specified with the format defined below. All of the watershed type

indices for one sector will fit on a single line. The watershed indices for the second and subsequent sectors are on the following lines, a new line for each sector in a clockwise rotation. A total of 16 lines are required. For example, a watershed index of 1 means that the water ingestion factor for watershed type 1 will be used for all material deposited on that spatial element. A watershed index of 2 means that the water ingestion factor for watershed type 2 will be used for all material deposited on that spatial element. The formatting of this information is illustrated below by the presentation of the watershed data for a single wind direction sector.

**WATERSHED INDEX**

1 1 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 3 3

The format used to process the data is as follows:

---

<u>Columns</u>	<u>Format</u>	<u>Variable</u>	<u>Identification</u>	<u>Allowed Range</u>
1-80	40I2	INDWTR	Watershed index	1 - NWTRSH

---

Agricultural Data Block

The length of the growing season and the average fraction of the farmland area at the site devoted to each crop type must be specified. These fractions, FRCLP, need not sum to exactly one, but their sum should not exceed a value of one. If these values sum to a value less than one, that sum indicates the fraction of farmland in production in an average year (some fraction of farmland may be fallow). Data must be given for each of the NCPGZN crop categories. The data block begins with the separator "CROP SEASON AND SHARE" in column 2. The formatting of this information is illustrated below by the presentation of the agricultural data for the seven crop categories used in the example Site Data File.

**CROP SEASON AND SHARE**

1 PASTURE	90.	270.	0.41
2 STORED FORAGE	150.	240.	0.13
3 GRAINS	150.	240.	0.21
4 GRN LEAFY VEGETABLES	150.	240.	0.002
5 OTHER FOOD CROPS	150.	240.	0.004
6 LEGUMES AND SEEDS	150.	240.	0.15
7 ROOTS AND TUBERS	150.	240.	0.003

The format used to process the data is as follows:

<u>Columns</u>	<u>Format</u>	<u>Variable</u>	<u>Identification</u>	<u>Allowed Range</u>
1-4	I4	I	Crop index	1 - NCPGZN
6-25	A20	CROP	Crop name	N.A.
26-30	F5.0	GBEG	Day of the year the growing season begins	1.0 - GEND
31-35	F5.0	GEND	Day of the year the growing season ends	GBEG - 365.0
36-45	F10.0	FRCLCP	Fraction of the site averaged farmland devoted to this crop	0.0 - 1.0

#### Watershed Data Block

The data block begins with the 22 character separator beginning with "WATERSHED DEFINITION" in column 2. For each of the NUMWPI (NWPIISO) nuclides considered in the liquid pathways model, an initial washoff fraction and an annual washoff rate (fraction/year) must be specified together with an ingestion factor (becquerel ingested/becquerel-in-water) for each of the NUMWPA (NWTRSH) watershed classes. The formatting of this information is illustrated below by the presentation of the watershed data for the four water ingestion nuclides used on the example Site Data File.

WATERSHED DEFINITION	RIVER	OCEAN	LAKE
1 SR-89	5.0E 6	0.0	2.0E 7
2 SR-90	5.0E 6	0.0	2.0E 7
3 CS-134	5.0E 6	0.0	2.0E 6
4 CS-137	5.0E 6	0.0	4.0E 6

The format used to process the data is as follows:

<u>Columns</u>	<u>Format</u>	<u>Variable</u>	<u>Identification</u>	<u>Allowed Range</u>
1-4	I4	I	Nuclide index	1 - NWPISO
6-13	A8	NMISO	Nuclide name	N.A.
16-25	E10.1	DUMMY	No longer used	N.A.
26-35	E10.1	DUMMY	No longer used	N.A.
36-45	E10.1	WTRINF(1)	Ingestion factor for watershed class 1	0.0 - 1.0
46-55	E10.1	WTRINF(2)	Ingestion factor for watershed class 2	0.0 - 1.0
56-65	E10.1	WTRINF(3)	Ingestion factor for watershed class 3	0.0 - 1.0
66-75	E10.1	WTRINF(4)	Ingestion factor for watershed class 4	0.0 - 1.0

#### Regional Economic Data Block

Economic data must be specified for each of the NECRGN economic regions. The data block begins with the separator "REGIONAL ECONOMIC DATA" in column 2. An economic region can be a county, a state or any other user-specified region. The data provided in this block are similar to those contained in the CRAC2 "Economic" subgroup but converted to metric units.

The index of the region is used with the "REGION INDEX" data block (described earlier in this Appendix) to associate each element of the spatial grid with an economic region. The formatting of this information is illustrated below by the presentation of the economic data for the first economic region on the example Site Data File, Alabama.

REGIONAL ECONOMIC DATA  
1 ALA .354 .040 459. 1824. 62000.

The format used to process this data is as follows:

<u>Columns</u>	<u>Format</u>	<u>Variable</u>	<u>Identification</u>	<u>Allowed Range</u>
1-4	I4	I	Region index number	1 - NECRGN
6-15	A10	NMRGN	Region identification	N.A.
21-25	F5.3	FRMFRC	Fraction of land devoted to farming in the region	0.0 - 1.0
26-30	F5.3	DPF	Fraction of farm sales resulting from dairy production in the region	0.0 - 1.0
31-40	F10.1	ASFP	Annual average farm sales for the region (\$/hectare)	0.0 - 1.E9
41-50	F10.1	VFRM	Average farmland value for the region (\$/hectare)	0.0 - 1.E9
51-60	F10.1	VNFRM	Average non farm value for the region (\$/person)	0.0 - 1.E9

## APPENDIX C — COUNTY DATABASE LISTING

This is a complete listing of the county-level census database. An explanation of the notes follows the last record of the database.

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1	ME	Androscoggin County	0 945790	0 206791	0 125332	3216	4532	109414	0
2	ME	Aroostook County	0 976975	0 078229	0 027361	908	2062	93633	0
3	ME	Cumberland County	0 686591	0 100777	0 233237	730	6779	144689	0
4	ME	Franklin County	0 973372	0 035752	0 595260	480	2296	101618	0
5	ME	Hancock County	0 675873	0 049238	0 013906	752	3909	126083	1
6	ME	Kennebec County	0 911939	0 171839	0 458911	947	3426	116032	0
7	ME	Knox County	0 320116	0 118053	0 223053	574	5220	117346	0
8	ME	Lincoln County	0 650885	0 083509	0 342677	482	7124	131854	0
9	ME	Oxford County	0 955295	0 047722	0 266111	565	3668	100968	0
10	ME	Penobscot County	0 954898	0 054361	0 513882	603	2904	105127	0
11	ME	Piscataquis County	0 906075	0 014176	0 601179	384	2186	93423	0
12	ME	Sagadahoc County	0 685980	0 115602	0 425645	372	3781	127058	0
13	ME	Somerset County	0 958764	0 042564	0 650222	452	2696	92833	0
14	ME	Waldo County	0 855772	0 153914	0 543369	589	3453	88803	0
15	ME	Washington County	0 789081	0 057641	0 003532	939	2675	91770	1
16	ME	York County	0 779422	0 097436	0 179135	626	5230	121294	0
17	NH	Belknap County	0 856434	0 081411	0 354882	351	6523	139609	0
18	NH	Carroll County	0 941140	0 042561	0 242416	339	4981	148692	0
19	NH	Cheshire County	0 970187	0 074948	0 286962	1342	8003	127275	0
20	NH	Coos County	0 983038	0 039965	0 801689	413	2442	113011	0
21	NH	Grafton County	0 978979	0 069059	0 713386	544	3933	130641	0
22	NH	Hillsborough County	0 982295	0 071030	0 292672	939	10445	152946	0
23	NH	Merrimack County	0 976960	0 077931	0 379773	936	5685	139528	0
24	NH	Rockingham County	0 875568	0 077069	0 229201	834	9055	147453	0
25	NH	Strafford County	0 960624	0 104706	0 278765	771	5980	116377	0
26	NH	Sullivan County	0 973679	0 111339	0 445552	822	5062	123049	0
27	VT	Addison County	0 952475	0 425459	0 795092	1103	3532	104605	0
28	VT	Bennington County	0 997935	0 077816	0 649364	490	4932	130391	0
29	VT	Caledonia County	0 989694	0 232118	0 817434	628	3299	96606	0
30	VT	Chittenden County	0 869815	0 240188	0 648925	635	5739	135058	0
31	VT	Essex County	0 987327	0 041594	0 835449	824	3122	80439	0
32	VT	Franklin County	0 919758	0 499110	0 854028	1143	3211	99939	0
33	VT	Grand Isle County	0 424469	0 469878	0 776269	872	4961	123116	0
34	VT	Lamoille County	0 993796	0 140267	0 759757	807	3755	109882	0
35	VT	Orange County	0 995328	0 211810	0 770961	655	3982	108283	0
36	VT	Orleans County	0 967437	0 335209	0 890911	906	3104	93043	0
37	VT	Rutland County	0 987034	0 222385	0 767982	509	3563	117339	0
38	VT	Washington County	0 991004	0 133442	0 735419	535	3981	120278	0
39	VT	Windham County	0 988228	0 087134	0 556936	791	6406	122981	0
40	VT	Windsor County	0 995116	0 144431	0 575939	358	4932	124031	0
41	MA	Barnstable County	0 303143	0 021083	0 014410	4026	27874	157592	1
42	MA	Berkshire County	0 984207	0 102299	0 543552	728	7194	135925	0
43	MA	Bristol County	0 804381	0 096202	0 198048	2138	18080	125446	0
44	MA	Dukes County	0 211446	0 086644	0 020024	364	23899	162672	0
45	MA	Essex County	0 601028	0 079902	0 127244	1724	28381	154659	0
46	MA	Franklin County	0.968735	0 165751	0 413335	996	7244	124539	1
47	MA	Hampden County	0 975318	0 094674	0 193698	1249	8728	129185	0
48	MA	Hampshire County	0 969861	0 157893	0 293337	1146	8717	120922	0
49	MA	Middlesex County	0 971587	0 059922	0 055635	3512	18182	171443	0
50	MA	Nantucket County	0 157553	0 020630	0 000000	9490	35197	199017	2
51	MA	Norfolk County	0 900073	0 038640	0 087926	2232	22306	182260	0
52	MA	Plymouth County	0 604114	0 170887	0 015790	3275	16090	145624	0
53	MA	Suffolk County	0 487363	0 002406	0 000000	9490	35197	156800	2
54	MA	Worcester County	0 958238	0 118548	0 287030	1068	9340	134753	0
55	RI	Bristol County	0 552055	0 092937	0 160382	3526	31856	148773	0
56	RI	Kent County	0 904617	0 051779	0 121551	1097	16907	136046	0
57	RI	Newport County	0 331750	0 144624	0 148933	3004	22232	137631	0
58	RI	Providence County	0 948249	0 046910	0 110992	1670	15397	121010	0
59	RI	Washington County	0 591389	0 096024	0 115392	1790	13975	131732	0
60	CT	Fairfield County	0 747724	0 024904	0 083056	2159	39050	219059	0
61	CT	Hartford County	0 979871	0 120050	0 043448	3972	20240	168483	0
62	CT	Litchfield County	0 973931	0 147046	0 500698	859	13066	170420	0
63	CT	Middlesex County	0 841054	0 083904	0 100521	2776	15377	163451	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
64	CT	New Haven County	0 702727	0 066757	0 127863	3072	26702	150345	0
65	CT	New London County	0 863116	0 154793	0 140664	3658	13471	137218	0
66	CT	Tolland County	0 983334	0 147512	0 638154	1368	12517	148631	0
67	CT	Windham County	0 983292	0 168386	0 520462	1510	10158	122392	0
68	NY	Albany County	0 981807	0 172694	0 002498	666	6622	142257	0
69	NY	Allegany County	0 995939	0 245144	0 006932	474	1711	85275	0
70	NY	Bronx County	0 731929	0 000000	0 000000	0	0	105520	0
71	NY	Broome County	0 987921	0 216337	0 004848	625	3004	122310	0
72	NY	Cattaraugus County	0 990631	0 242979	0 006846	680	2684	92779	0
73	NY	Cayuga County	0 890032	0 572469	0 002607	914	3203	100914	0
74	NY	Chautauqua County	0 708007	0 381804	0 004889	841	3210	102932	0
75	NY	Chemung County	0 993630	0 225703	0 004951	499	2361	107592	0
76	NY	Chenango County	0 995171	0 328439	0 006224	735	2583	98916	0
77	NY	Clinton County	0 929966	0 238104	0 004210	889	2748	92739	0
78	NY	Columbia County	0 980669	0 275184	0 001940	1228	7247	133080	0
79	NY	Cortland County	0 996270	0 433462	0 005138	770	2839	95936	0
80	NY	Delaware County	0 985239	0 207531	0 006333	658	3041	96396	0
81	NY	Dutchess County	0 971196	0 213800	0 002025	745	11073	150182	0
82	NY	Erie County	0 851419	0 217891	0 003145	1203	5177	123983	0
83	NY	Essex County	0 937565	0 047812	0 004305	376	3360	102465	0
84	NY	Franklin County	0 961161	0 132441	0 006241	853	2411	94147	0
85	NY	Fulton County	0 931021	0 111298	0 006473	713	3369	106238	0
86	NY	Genesee County	0 997511	0 543003	0 001873	1022	3145	116777	0
87	NY	Greene County	0 984297	0 110510	0 004196	437	6988	111141	0
88	NY	Hamilton County	0 951760	0 000015	0 000000	7562	2288	112510	3
89	NY	Herkimer County	0 968037	0 180473	0 006572	768	2529	97256	0
90	NY	Jefferson County	0 685021	0 369106	0 005745	627	2027	102986	0
91	NY	Kings County	0 728098	0 000089	0 000000	85869	244261	113810	4
92	NY	Lewis County	0 988826	0 207398	0 006695	911	2625	84780	0
93	NY	Livingston County	0 987016	0 506944	0 001987	740	3012	109577	0
94	NY	Madison County	0 991404	0 466044	0 005285	858	2878	109049	0
95	NY	Monroe County	0 482662	0 261052	0 001205	931	6070	146579	0
96	NY	Montgomery County	0 986542	0 535809	0 005736	881	2993	108872	0
97	NY	Nassau County	0 632598	0 010298	0 000000	2663	64475	214568	0
98	NY	New York County	0 842529	0 000000	0 000000	0	0	262759	0
99	NY	Niagara County	0 458790	0 404821	0 001499	839	3451	112753	0
100	NY	Oneida County	0 964676	0 312610	0 006033	765	2971	111602	0
101	NY	Onondaga County	0 968467	0 291002	0 002948	1055	4116	131407	0
102	NY	Ontario County	0 972754	0 440384	0 002180	767	4167	124972	0
103	NY	Orange County	0 973522	0 196626	0 001795	1795	10604	134028	0
104	NY	Orleans County	0 478793	0 534321	0 000989	952	2927	104748	0
105	NY	Oswego County	0 726502	0 184115	0 004570	688	3246	100474	0
106	NY	Otsego County	0 987878	0 340107	0 006748	659	2972	105005	0
107	NY	Putnam County	0 940190	0 025664	0 000000	1251	16855	182111	0
108	NY	Queens County	0 616847	0 000474	0 000000	7562	244261	140239	5
109	NY	Rensselaer County	0 982830	0 221423	0 004105	702	4071	120387	0
110	NY	Richmond County	0 571124	0 000885	0 000000	7562	244261	158811	5
111	NY	Rockland County	0 873939	0 007453	0 000000	4333	129699	181230	0
112	NY	St. Lawrence County	0 951839	0.230804	0 007019	587	1932	86047	0
113	NY	Saratoga County	0 962231	0 135491	0 003191	836	5514	127722	0
114	NY	Schenectady County	0 983231	0 145527	0 000682	1509	5469	140063	1
115	NY	Schoharie County	0 993066	0 295995	0 005793	623	3564	96674	0
116	NY	Schuyler County	0 949640	0 310496	0 004412	489	2421	86677	0
117	NY	Seneca County	0 832015	0.553356	0 002491	681	3309	108439	0
118	NY	Steuben County	0 991849	0 407579	0 005079	560	2276	100027	0
119	NY	Suffolk County	0 383937	0 060620	0 000000	9349	26910	164331	0
120	NY	Sullivan County	0 972789	0 090228	0 003957	903	6869	123922	0
121	NY	Tioga County	0 991946	0 345978	0 005437	665	2727	103440	0
122	NY	Tompkins County	0 968321	0 301362	0 002070	1365	3812	103434	0
123	NY	Ulster County	0 970504	0 096591	0 000530	1806	8135	127499	0
124	NY	Warren County	0 933450	0 010440	0 000656	648	6953	117224	1
125	NY	Washington County	0 987732	0 385146	0 004086	975	3519	94669	0
126	NY	Wayne County	0 436523	0 451575	0 001288	1275	3965	115998	0
127	NY	Westchester County	0 865412	0 020607	0 000000	3083	30519	225751	0
128	NY	Wyoming County	0 994126	0 553057	0 003342	1237	3207	90002	0
129	NY	Yates County	0 909635	0 471286	0 004848	789	3228	88600	0
130	NJ	Atlantic County	0 835706	0 082436	0 000000	3626	12307	162794	0
131	NJ	Bergen County	0 948889	0 017586	0 000000	6211	84180	218300	0
132	NJ	Burlington County	0 982049	0 188692	0 096929	1642	12922	155838	0
133	NJ	Camden County	0 976911	0 054810	0 000000	2592	19966	134990	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
134	NJ	Cape May County	0 411443	0 071288	0 000000	1191	9959	152566	0
135	NJ	Cumberland County	0 723235	0 219141	0 024742	2628	9425	113918	0
136	NJ	Essex County	0 974518	0 007584	0 000000	3680	75792	166099	0
137	NJ	Gloucester County	0 963994	0 296989	0 042639	2184	13895	125460	0
138	NJ	Hudson County	0 749307	0 000000	0 000000	0	0	125054	0
139	NJ	Hunterdon County	0 982287	0 386267	0 190300	609	22180	204923	0
140	NJ	Mercer County	0 987260	0 247461	0 053656	1096	27535	176279	0
141	NJ	Middlesex County	0 962761	0 125808	0 011310	2324	24824	168626	0
142	NJ	Monmouth County	0 709208	0 194567	0 013210	2142	31841	185525	0
143	NJ	Morris County	0 974473	0 079661	0 018392	1825	30889	213491	1
144	NJ	Ocean County	0 694776	0 025450	0 069560	1203	20783	142061	0
145	NJ	Passaic County	0 939024	0 015519	0 023743	4598	31672	148827	1
146	NJ	Salem County	0 906869	0 454421	0 160871	1369	8249	120800	0
147	NJ	Somerset County	0 998871	0 225556	0 161104	710	19066	224735	0
148	NJ	Sussex County	0 972542	0 226416	0 440014	647	13969	165300	0
149	NJ	Union County	0 979385	0 004916	0 000000	50455	148654	177783	0
150	NJ	Warren County	0 986595	0 382590	0 264670	1126	16315	146680	0
151	PA	Adams County	0 997114	0 517874	0 143371	1773	6612	113844	0
152	PA	Allegheny County	0 980510	0 069599	0 053454	866	7340	140077	0
153	PA	Armstrong County	0 984185	0 285673	0 111610	1404	4641	113370	0
154	PA	Beaver County	0 980461	0 208063	0 464782	525	5487	102695	0
155	PA	Bedford County	0 997157	0 306550	0 693447	675	3727	87537	0
156	PA	Berks County	0 992537	0 403675	0 218360	2646	9077	133127	0
157	PA	Blair County	0 997511	0 227206	0 713114	1348	5322	100101	0
158	PA	Bradford County	0 991087	0 421835	0 472403	1031	3234	103264	0
159	PA	Bucks County	0 976689	0 197458	0 142950	1989	14091	152722	0
160	PA	Butler County	0 992121	0 256239	0 359425	540	5237	116844	0
161	PA	Cambria County	0 992227	0 174844	0 271307	536	3556	98029	0
162	PA	Cameron County	0 996401	0 013036	0 861328	191	4417	100921	6
163	PA	Carbon County	0 984001	0 077707	0 151429	686	7885	102797	0
164	PA	Centre County	0 996117	0 197381	0 611454	791	6274	104822	0
165	PA	Chester County	0 994941	0 365079	0 161821	3953	14121	167515	0
166	PA	Clarion County	0 989315	0 245900	0 623767	362	2750	94750	0
167	PA	Clearfield County	0 994375	0 074930	0 518523	377	3046	97636	0
168	PA	Clinton County	0 991952	0 069123	0 516480	1071	5011	88831	0
169	PA	Columbia County	0 991350	0 327629	0 246305	691	5092	97575	0
170	PA	Crawford County	0 975989	0 325545	0 649485	619	2749	97649	0
171	PA	Cumberland County	0 998157	0 403046	0 507543	1317	7514	130411	0
172	PA	Dauphin County	0 942166	0 268578	0 265823	1352	8348	124275	0
173	PA	Delaware County	0 965714	0 043213	0 031759	3367	36916	160240	1
174	PA	Elk County	0 995700	0 030868	0 353056	464	3926	114277	0
175	PA	Erie County	0 514596	0 327042	0 353668	956	4835	110613	0
176	PA	Fayette County	0 990135	0 210390	0 412694	394	3718	95170	0
177	PA	Forest County	0 992398	0 017160	0 451724	305	16890	88831	7
178	PA	Franklin County	0 999071	0 474413	0 533344	1777	6498	110274	0
179	PA	Fulton County	0 998941	0 317723	0 513678	552	2906	97487	0
180	PA	Greene County	0 996402	0 341056	0 331161	133	2012	85647	0
181	PA	Huntingdon County	0 983232	0 231158	0 661991	698	3523	82349	0
182	PA	Indiana County	0 994222	0 269423	0 331664	714	3435	93748	0
183	PA	Jefferson County	0 997864	0 189047	0 445880	463	2957	100304	0
184	PA	Juniata County	0 994972	0 339580	0 311113	1481	6054	99065	0
185	PA	Lackawanna County	0 987555	0 125884	0 324010	765	5917	114894	0
186	PA	Lancaster County	0 964660	0 639348	0 288362	4332	13630	127864	0
187	PA	Lawrence County	0 993697	0 374499	0 535546	686	3940	100067	0
188	PA	Lebanon County	0 997807	0 451369	0 333374	3117	9770	111758	0
189	PA	Lehigh County	0 995235	0 374012	0 071928	1270	10369	134008	0
190	PA	Luzerne County	0 982155	0 087424	0 256540	735	6422	111114	0
191	PA	Lycoming County	0 992834	0 168278	0 376052	722	4282	106868	0
192	PA	Mc Kean County	0 997356	0 062972	0 581512	237	2651	97588	0
193	PA	Mercer County	0 984242	0 373964	0 528594	631	3121	102797	0
194	PA	Mifflin County	0 993358	0 309773	0 540079	1417	4737	91499	0
195	PA	Monroe County	0 985864	0 053453	0 183507	480	7358	119154	0
196	PA	Montgomery County	0 991118	0 143672	0 202605	1542	16576	192779	0
197	PA	Montour County	0 988298	0 494087	0 366617	755	5852	122703	0
198	PA	Northampton County	0 990569	0 340538	0 310226	890	8579	124193	0
199	PA	Northumberland County	0 963407	0 371789	0 155106	1158	5087	98686	0
200	PA	Perry County	0 995957	0 294376	0 383693	960	5257	101198	0
201	PA	Philadelphia County	0 947114	0 009196	0 000000	1588	16945	113255	8
202	PA	Pike County	0 965437	0 017699	0 064532	834	7709	129673	0
203	PA	Potter County	0 999779	0 130155	0 622430	519	2892	94127	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
204	PA	Schuylkill County	0 994854	0 178695	0 167082	1181	6041	107376	0
205	PA	Snyder County	0 997137	0 411600	0 278921	1574	5597	119276	0
206	PA	Somerset County	0 993985	0 319744	0 673550	644	3184	96654	0
207	PA	Sullivan County	0 994707	0 106300	0 569905	636	3265	97094	0
208	PA	Susquehanna County	0 988667	0 336448	0 700473	587	3173	94737	0
209	PA	Tioga County	0 996853	0 292823	0 695515	551	2658	88004	0
210	PA	Union County	0 998769	0 311553	0 375789	1463	7467	103610	0
211	PA	Venango County	0 988335	0 122110	0 505558	299	2780	110180	0
212	PA	Warren County	0 984001	0 119182	0 711556	551	2384	114589	0
213	PA	Washington County	0 995521	0 370100	0 414700	328	4420	112604	0
214	PA	Wayne County	0 971720	0 261159	0 755978	530	5310	110037	0
215	PA	Westmoreland County	0 986604	0 235156	0 361035	620	5011	115612	0
216	PA	Wyoming County	0 981238	0 246789	0 314229	1109	3721	105005	0
217	PA	York County	0 993731	0 435350	0 212071	1177	7821	132091	0
218	OH	Adams County	0 996796	0 525421	0 170948	330	2839	68159	0
219	OH	Allen County	0 993965	0 749540	0 043569	662	4582	112042	0
220	OH	Ashland County	0 994207	0 630055	0 425041	673	3856	101463	0
221	OH	Ashtabula County	0 513388	0 346641	0 476565	577	3360	90822	0
222	OH	Athens County	0 996496	0 248211	0 418204	178	2427	78474	0
223	OH	Auglaize County	0 998838	0 798652	0 202421	824	4397	109523	0
224	OH	Belmont County	0 992421	0 366979	0 423841	216	2285	95949	0
225	OH	Brown County	0 993020	0 642373	0 069776	425	3528	88126	0
226	OH	Butler County	0 993768	0 462412	0 132490	616	5696	113451	0
227	OH	Carroll County	0 989135	0 484886	0 505121	400	3027	88275	0
228	OH	Champaign County	0 997332	0 786713	0 133267	670	3901	96938	0
229	OH	Clark County	0 990581	0 700269	0 024231	886	4704	112950	0
230	OH	Clermont County	0 987582	0 342927	0 050908	457	6000	109936	0
231	OH	Clinton County	0 996589	0 864648	0 014912	626	4346	109103	0
232	OH	Columbiana County	0 994972	0 421845	0 431187	759	3910	85322	0
233	OH	Coshcohton County	0 993877	0 447932	0 283061	426	3030	92969	0
234	OH	Crawford County	0 998711	0 866891	0 062250	616	3712	94249	0
235	OH	Cuyahoga County	0 367913	0 013843	0 000000	9687	26979	142820	0
236	OH	Darke County	0 999230	0 874012	0 080047	1467	5392	109177	0
237	OH	Defiance County	0 992779	0 747626	0 084824	498	3340	104361	0
238	OH	Delaware County	0 970497	0 596758	0 060701	596	6466	128549	0
239	OH	Erie County	0 406691	0 545797	0 103452	725	5392	116093	0
240	OH	Fairfield County	0 994406	0 610913	0 073027	543	4517	110688	0
241	OH	Fayette County	0 998779	0 906776	0 008709	616	3536	93192	0
242	OH	Franklin County	0 993779	0 280746	0 048271	1008	7469	128962	0
243	OH	Fulton County	0 998678	0 789785	0 038477	1003	5098	113769	0
244	OH	Gallia County	0 994985	0 366030	0 239265	310	2583	86914	0
245	OH	Geauga County	0 988813	0 252333	0 474761	609	7783	140320	0
246	OH	Greene County	0 996823	0 693966	0 031290	688	5108	123008	0
247	OH	Guernsey County	0 988106	0 382712	0 279282	186	2381	86975	0
248	OH	Hamilton County	0 986874	0 111160	0 053501	1363	9286	139392	0
249	OH	Hancock County	0 995798	0 810499	0 031032	562	4857	127743	0
250	OH	Hardin County	0 999479	0 825261	0 041118	539	3287	86751	0
251	OH	Harrison County	0 982362	0 438779	0 417491	183	1808	73496	0
252	OH	Henry County	0 991736	0 919228	0 043931	646	4439	109523	0
253	OH	Highland County	0 991826	0 652271	0 075005	473	3185	89779	0
254	OH	Hocking County	0 998019	0 177583	0 000000	136	2729	83541	0
255	OH	Holmes County	0 997574	0 654511	0 372779	1101	4652	74614	0
256	OH	Huron County	0 996292	0 694017	0 094496	717	3970	111616	0
257	OH	Jackson County	0 997106	0 276815	0 047491	617	2379	77838	0
258	OH	Jefferson County	0 996941	0 260689	0 356231	232	1952	98774	0
259	OH	Knox County	0 995449	0 624229	0 185857	680	3294	101225	0
260	OH	Lake County	0 233123	0 117328	0 002976	7176	12755	133378	1
261	OH	Lawrence County	0 995947	0 212698	0 186066	191	2918	80201	0
262	OH	Licking County	0 997275	0 517394	0 086666	1023	5129	111162	0
263	OH	Logan County	0 982148	0 691581	0 148281	681	3445	103942	0
264	OH	Lorain County	0 533613	0 452421	0 131790	1223	5722	108412	0
265	OH	Lucas County	0 607153	0 339846	0 002428	1222	6301	120665	1
266	OH	Madison County	0 998448	0 877624	0 045349	638	4087	105716	0
267	OH	Mahoning County	0 980798	0 276491	0 322282	934	5444	110105	0
268	OH	Marion County	0 999248	0 841086	0 037738	563	3547	91777	0
269	OH	Medina County	0 996284	0 384945	0 350805	653	6249	127959	0
270	OH	Meigs County	0 993165	0 316667	0 273898	293	2356	78312	0
271	OH	Mercer County	0 978809	0 907792	0 179281	1751	5435	114636	0
272	OH	Miami County	0 994721	0 769314	0 066317	767	5113	114975	0
273	OH	Monroe County	0 995810	0 377235	0 380866	186	2123	89352	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
274	OH	Montgomery County	0 994220	0 362640	0 053411	724	6973	124695	0
275	OH	Morgan County	0 990082	0 426042	0 191419	231	2006	87903	0
276	OH	Morrow County	0 995501	0 634204	0 141746	533	3390	89325	0
277	OH	Muskingum County	0 988158	0 441303	0 154155	334	2626	101347	0
278	OH	Noble County	0 986197	0 408020	0 129822	114	2159	84035	0
279	OH	Ottawa County	0 411174	0 652775	0 045275	551	4469	124810	0
280	OH	Paulding County	0 993703	0 822151	0 025545	620	3672	100501	0
281	OH	Perry County	0 993746	0 366051	0 134772	364	3110	79165	0
282	OH	Pickaway County	0 990754	0 807111	0 036354	557	3761	93186	0
283	OH	Pike County	0 994466	0 311262	0 149070	263	2629	81366	0
284	OH	Portage County	0 970976	0 305549	0 290331	600	5686	108581	0
285	OH	Preble County	0 996432	0 750590	0 073311	748	4542	100900	0
286	OH	Putnam County	0 999248	0 925718	0 052218	753	4948	110369	0
287	OH	Richland County	0 993358	0 505288	0 227102	680	4262	109665	0
288	OH	Ross County	0 993439	0 575073	0 045470	428	2964	91276	0
289	OH	Sandusky County	0 979429	0 772552	0 039436	672	4265	115138	0
290	OH	Scioto County	0 993839	0 244213	0 106403	420	3321	85844	0
291	OH	Seneca County	0 996777	0 840999	0 037909	612	3605	100650	0
292	OH	Shelby County	0 995721	0 776844	0 235278	749	4802	118917	0
293	OH	Stark County	0 991798	0 370471	0 327651	1021	5537	114454	0
294	OH	Summit County	0 982640	0 072250	0 036828	1009	11165	125866	0
295	OH	Trumbull County	0 971235	0 305779	0 412657	515	4554	109347	0
296	OH	Tuscarawas County	0 993176	0 408717	0 435107	770	3813	101314	0
297	OH	Union County	0 999255	0 797774	0 060319	862	3845	130682	0
298	OH	Van Wert County	0 999026	0 921196	0 027532	609	4760	102682	0
299	OH	Vinton County	0 997880	0 157213	0 129919	125	2219	74830	0
300	OH	Warren County	0 982155	0 505688	0 103877	610	8021	116689	0
301	OH	Washington County	0 992281	0 343507	0 257153	339	2600	99193	0
302	OH	Wayne County	0 998305	0 694708	0 468025	1391	5264	111514	0
303	OH	Williams County	0 996965	0 693372	0 051111	635	3097	113295	0
304	OH	Wood County	0 994837	0 765472	0 012281	675	4518	118680	0
305	OH	Wyandot County	0 995073	0 833244	0 051345	568	3463	103989	0
306	IN	Adams County	0 998308	0 910339	0 142896	1008	5134	98611	0
307	IN	Allen County	0 995679	0 679239	0 068076	690	5410	129524	0
308	IN	Bartholomew County	0 993843	0 634016	0 029805	669	4584	116594	0
309	IN	Benton County	0 999776	1 040629	0 003650	575	4625	114535	9
310	IN	Blackford County	0 998212	0 826373	0 010354	642	2765	101754	0
311	IN	Boone County	0 998674	0 825809	0 020779	805	5067	148936	0
312	IN	Brown County	0 986237	0 112850	0 059013	260	3079	88560	1
313	IN	Carroll County	0 992541	0 923636	0 004832	1160	4978	112821	0
314	IN	Cass County	0 995046	0 861727	0 041420	780	4161	110559	0
315	IN	Clark County	0 997170	0 440046	0 095021	415	3684	105709	0
316	IN	Clay County	0 992309	0 709645	0 044909	577	3289	95888	0
317	IN	Clinton County	0 999560	0 910495	0 000673	902	4455	110803	0
318	IN	Crawford County	0 989741	0 305312	0 041180	588	2543	78115	0
319	IN	Daviess County	0 985946	0 806867	0 039309	1179	4049	95841	0
320	IN	Dearborn County	0 994014	0 441430	0 116576	264	4058	103237	0
321	IN	Decatur County	0 997822	0 848816	0 037442	905	4402	107694	0
322	IN	De Kalb County	0 997357	0 659663	0 179433	541	3315	106082	0
323	IN	Delaware County	0 993351	0 672434	0 028253	626	3788	106319	0
324	IN	Dubois County	0 988195	0 702511	0 045328	1768	3870	126876	0
325	IN	Elkhart County	0 991389	0 647824	0 346297	1245	5854	113587	0
326	IN	Fayette County	0 999282	0 810372	0 037704	611	3457	101259	0
327	IN	Floyd County	0 997889	0 314964	0 051388	340	5200	116235	0
328	IN	Fountain County	0 994406	0 904577	0 003971	543	3680	100961	0
329	IN	Franklin County	0 986374	0 601729	0 118187	576	3489	98435	0
330	IN	Fulton County	0 992282	0 823858	0 089932	631	3224	100948	0
331	IN	Gibson County	0 979516	0 770453	0 029507	676	4303	109787	0
332	IN	Grant County	0 998054	0 741696	0 032807	716	4332	107071	0
333	IN	Greene County	0 993018	0 598805	0 055258	488	3222	90686	0
334	IN	Hamilton County	0 988256	0 638596	0 019221	763	5834	173943	0
335	IN	Hancock County	0 998133	0 833083	0 002709	698	4991	128013	0
336	IN	Harrison County	0 996589	0 520815	0 084954	695	3586	96173	0
337	IN	Hendricks County	0 998828	0 715702	0 021472	639	4929	128935	0
338	IN	Henry County	0 995009	0 758666	0 025667	690	3597	105506	0
339	IN	Howard County	0 997072	0 792270	0 025360	938	5371	120163	0
340	IN	Huntington County	0 986215	0 767575	0 042786	690	4475	110566	0
341	IN	Jackson County	0 991360	0 622413	0 056330	895	3946	104659	0
342	IN	Jasper County	0 997582	0 842677	0 004232	802	3439	111358	0
343	IN	Jay County	0 999531	0 744619	0 074106	864	3509	96342	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
344	IN	Jefferson County	0 995660	0 565635	0 078511	428	3061	89007	0
345	IN	Jennings County	0 997021	0 516458	0 022858	705	3517	89596	0
346	IN	Johnson County	0 995775	0 681354	0 051447	763	4931	127025	0
347	IN	Knox County	0 984307	0 925743	0 011510	803	3430	98029	0
348	IN	Kosciusko County	0 969621	0 731378	0 070177	1232	4058	122039	0
349	IN	Lagrange County	0 981501	0 772030	0 167248	1295	4650	90199	0
350	IN	Lake County	0 793489	0 453653	0 062139	575	4206	110586	0
351	IN	La Porte County	0 975843	0 699130	0 131069	707	3796	112808	0
352	IN	Lawrence County	0 992846	0 552746	0 062329	267	2451	98950	0
353	IN	Madison County	0 998277	0 771733	0 014727	727	4642	108493	0
354	IN	Marion County	0 983374	0 152868	0 013547	1319	8072	129693	1
355	IN	Marshall County	0 987444	0 771580	0 197856	655	3398	104842	0
356	IN	Martin County	0 987101	0 332786	0 008148	983	3051	88634	0
357	IN	Miami County	0 995709	0 785249	0 083955	846	4043	96945	0
358	IN	Monroe County	0 958752	0 234873	0 059757	428	4190	96626	0
359	IN	Montgomery County	0 998402	0 875608	0 003662	799	3807	121836	0
360	IN	Morgan County	0 992879	0 536304	0 016435	541	4070	104659	0
361	IN	Newton County	0 995826	0 804364	0 002098	799	3719	102262	1
362	IN	Noble County	0 984438	0 699736	0 161210	662	3367	100277	0
363	IN	Ohio County	0 991464	0 582219	0 095728	336	3126	91506	1
364	IN	Orange County	0 978734	0 453900	0 061895	425	2426	87408	0
365	IN	Owen County	0 993198	0 458871	0 053295	300	3016	90415	0
366	IN	Parke County	0 988232	0 638084	0 023882	489	3142	100690	0
367	IN	Perry County	0 987194	0 328048	0 073466	342	3094	89061	0
368	IN	Pike County	0 985614	0 396750	0 013925	583	2601	102438	1
369	IN	Porter County	0 801732	0 532324	0 036438	602	4495	118978	0
370	IN	Posey County	0 973902	0 845117	0 036221	696	3971	105574	0
371	IN	Pulaski County	0 997941	0 874659	0 034842	766	3579	114894	0
372	IN	Putnam County	0 995289	0 664127	0 038882	536	3374	97033	0
373	IN	Randolph County	0 999196	0 815700	0.033009	689	3241	89704	0
374	IN	Ripley County	0 996532	0 574124	0 078734	564	3294	105987	0
375	IN	Rush County	0 999252	0 892253	0 046845	783	4168	105330	0
376	IN	St Joseph County	0 992120	0 588827	0 101528	714	4737	116574	0
377	IN	Scott County	0 987837	0 519729	0 051243	388	2853	82152	0
378	IN	Shelby County	0 998858	0 822727	0 034493	720	4710	113661	0
379	IN	Spencer County	0 993670	0 686279	0 049181	644	3375	98821	0
380	IN	Starke County	0 990385	0 681732	0 011687	519	2798	81698	0
381	IN	Steuben County	0 957332	0 615970	0 245508	508	3120	120963	0
382	IN	Sullivan County	0 984913	0 632442	0 006032	588	3440	101869	0
383	IN	Switzerland County	0 989531	0 559692	0 162546	411	3821	72419	0
384	IN	Tippecanoe County	0 993483	0 804483	0 010542	696	4684	107335	0
385	IN	Tipton County	0 999913	0 965618	0 004813	895	5588	122676	1
386	IN	Union County	0 977609	0 774380	0 043556	726	4552	94974	0
387	IN	Vanderburgh County	0 995039	0 539235	0 042686	680	4905	122547	0
388	IN	Vermillion County	0 988277	0 725698	0 002290	561	3378	97988	0
389	IN	Vigo County	0 982524	0 560661	0 020682	518	3368	101842	0
390	IN	Wabash County	0 981233	0 748539	0 038606	1213	4210	104449	0
391	IN	Warren County	0 995315	0 863827	0 002235	532	3691	107057	0
392	IN	Warrick County	0 982641	0 391426	0 037880	508	4081	123130	0
393	IN	Washington County	0 995904	0 574442	0 139293	569	2821	84821	0
394	IN	Wayne County	0 998019	0 733548	0 067080	763	3142	103772	0
395	IN	Wells County	0 998795	0 839050	0 058166	688	4348	109570	0
396	IN	White County	0 992993	0 881878	0 009089	907	4336	109421	0
397	IN	Whitley County	0 992956	0 755536	0 098252	683	3623	110200	0
398	IL	Adams County	0 983221	0 847790	0 051900	585	2975	113214	0
399	IL	Alexander County	0 936044	0 458399	0 009069	487	2512	79646	0
400	IL	Bond County	0 993565	0 750294	0 127589	591	3239	107660	0
401	IL	Boone County	0 997976	0 750532	0 159228	738	5511	117380	0
402	IL	Brown County	0 994823	0 738308	0 003311	457	2641	93524	1
403	IL	Bureau County	0 994577	0 867337	0 005796	867	4555	113234	0
404	IL	Calhoun County	0 894759	0 613524	0 006035	435	2825	104111	0
405	IL	Carroll County	0 953720	0 840274	0 061224	1026	4158	101422	0
406	IL	Cass County	0 979680	0 870424	0 000000	730	3960	102661	0
407	IL	Champaign County	0 999630	0 895961	0 004417	692	5530	114853	0
408	IL	Christian County	0 990704	0 859743	0 000000	749	5315	116194	0
409	IL	Clark County	0 993311	0 809771	0 004596	548	3582	98848	1
410	IL	Clay County	0 998743	0 745040	0 004762	410	2682	102831	1
411	IL	Clinton County	0 941918	0 754860	0 275238	1025	3951	115016	0
412	IL	Coles County	0 996511	0 809732	0 007795	610	4860	103014	0
413	IL	Cook County	0 578432	0 067606	0 007683	1234	14850	147175	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
414	IL	Crawford County	0 995054	0 787517	0 008232	585	3401	108954	0
415	IL	Cumberland County	0 997175	0 794764	0 098696	714	3826	91770	0
416	IL	DeKalb County	0 998682	0 930116	0 015321	1033	6199	111460	0
417	IL	De Witt County	0 981295	0 810615	0 001607	659	5734	122818	1
418	IL	Douglas County	0 998662	0 972610	0 030653	755	5215	98801	0
419	IL	DuPage County	0 992783	0 085058	0 004343	2762	8382	181332	1
420	IL	Edgar County	0 998991	0 888209	0 002877	642	4551	105330	1
421	IL	Edwards County	0 998642	0 817289	0 004202	551	2437	104578	0
422	IL	Effingham County	0 997494	0 841293	0 147956	684	4177	118599	0
423	IL	Fayette County	0 987772	0 744204	0 053737	512	2777	87584	0
424	IL	Ford County	0 998951	0 965063	0 003394	643	4457	116533	1
425	IL	Franklin County	0 955203	0 608672	0 014432	396	2818	94127	0
426	IL	Fulton County	0 980773	0 778667	0 003726	490	2992	96362	0
427	IL	Gallatin County	0 985754	0 829839	0 004690	542	3021	105215	1
428	IL	Greene County	0 994056	0 873747	0 007300	557	2946	90652	0
429	IL	Grundy County	0 976063	0 838660	0 003339	580	6156	132714	1
430	IL	Hamilton County	0 998453	0 723702	0 002718	399	2189	91079	1
431	IL	Hancock County	0 975551	0 851864	0 004089	598	3228	99898	0
432	IL	Hardin County	0 982375	0 332715	0 000000	133	1684	77350	0
433	IL	Henderson County	0 958716	0 841283	0 005091	631	3515	100934	1
434	IL	Henry County	0 997112	0 861562	0 003127	930	4358	108039	0
435	IL	Iroquois County	0 998566	0 927318	0 018908	701	4227	104571	0
436	IL	Jackson County	0 976101	0 495270	0 025206	422	3056	87584	0
437	IL	Jasper County	0 992739	0 815360	0 052370	668	3674	106908	0
438	IL	Jefferson County	0 978300	0 594253	0 025796	341	2522	110749	0
439	IL	Jersey County	0 979265	0 764573	0 035073	632	3451	102634	0
440	IL	Jo Daviess County	0 971528	0 754916	0 321967	649	3146	112801	0
441	IL	Johnson County	0 991838	0 427523	0 027205	278	1985	68247	0
442	IL	Kane County	0 993480	0 610926	0 055233	1038	10008	146870	0
443	IL	Kankakee County	0 994158	0 827759	0 014771	724	5224	110288	0
444	IL	Kendall County	0 993692	0 868446	0 009765	679	7163	136561	0
445	IL	Knox County	0 995251	0 841019	0 008128	698	3925	105716	0
446	IL	Lake County	0 327316	0 255222	0 035874	1073	8794	185437	0
447	IL	La Salle County	0 988613	0 842633	0 006246	638	5688	109177	0
448	IL	Lawrence County	0 994757	0 711063	0 019004	643	3237	107707	0
449	IL	Lee County	0 994617	0 892688	0 012731	750	4918	113153	0
450	IL	Livingston County	0 998400	0 954352	0 022391	679	4524	113187	0
451	IL	Logan County	0 998602	0 935095	0 000865	733	5525	110044	0
452	IL	McDonough County	0 998727	0 913798	0 001342	620	3863	88614	0
453	IL	McHenry County	0 988494	0 644607	0 124103	945	7112	148028	0
454	IL	McLean County	0 997733	0 936076	0 008072	691	5707	128427	0
455	IL	Macon County	0 991719	0 835724	0.000958	734	5620	122060	1
456	IL	Macoupin County	0 995420	0 727830	0 025287	712	3731	108242	0
457	IL	Madison County	0 979306	0 645867	0 092233	689	4859	121782	0
458	IL	Marion County	0 994021	0 693261	0 028101	415	2394	101239	0
459	IL	Marshall County	0 968828	0 824523	0 009394	608	5155	114806	0
460	IL	Mason County	0 956644	0 818156	0 002425	638	3745	103908	1
461	IL	Massac County	0 987297	0 646025	0 016466	403	2184	89596	1
462	IL	Menard County	0 996429	0 816163	0 001860	715	4510	118849	1
463	IL	Mercer County	0 986186	0 869263	0 007130	650	3074	105940	0
464	IL	Monroe County	0 976389	0 752568	0 052782	545	4051	140063	0
465	IL	Montgomery County	0 991632	0 825675	0 022856	651	3598	102810	0
466	IL	Morgan County	0 993919	0 855033	0 004116	675	4592	116797	0
467	IL	Moultrie County	0 974245	0 859442	0 029346	746	5894	104057	0
468	IL	Ogle County	0 994190	0 808406	0 035609	832	4529	108439	0
469	IL	Peoria County	0 981991	0 659454	0 017239	604	4784	124234	0
470	IL	Perry County	0 986949	0 593818	0 043696	342	2215	100677	0
471	IL	Piatt County	0 999289	0 892253	0 004939	691	5386	123672	0
472	IL	Pike County	0 978117	0 834513	0 004873	542	2654	96897	0
473	IL	Pope County	0 989925	0 286474	0 000000	180	2218	70536	0
474	IL	Pulaski County	0 987561	0 641343	0 024739	468	2209	91181	0
475	IL	Putnam County	0.927744	0 763449	0 007457	1035	4600	124071	0
476	IL	Randolph County	0 968450	0 730935	0 100583	411	2807	101307	0
477	IL	Richland County	0 994841	0 819930	0 021377	596	3298	106854	0
478	IL	Rock Island County	0 945900	0 643800	0 008049	735	4118	120685	0
479	IL	St. Clair County	0 985032	0 621659	0 032770	614	4805	109658	0
480	IL	Saline County	0 990532	0 577603	0 003111	471	2322	111778	0
481	IL	Sangamon County	0 989926	0 803946	0 001074	762	5532	132761	0
482	IL	Schuylerville County	0 990776	0 740916	0 005134	399	2115	96979	0
483	IL	Scott County	0 992765	0 802338	0 000000	592	3408	112537	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
484	IL	Shelby County	0 987584	0 828495	0 042253	630	4067	106570	0
485	IL	Stark County	0 998985	0 920471	0 004353	679	5085	115551	0
486	IL	Stephenson County	0 999107	0 871927	0 357337	1019	4012	122744	0
487	IL	Tazewell County	0 986282	0 810131	0 016924	759	5724	121972	0
488	IL	Union County	0 985875	0 448158	0 042443	408	2281	96809	0
489	IL	Vermilion County	0 996614	0 848423	0 003622	632	4431	100379	0
490	IL	Wabash County	0 981156	0 807639	0 028910	565	3361	116499	0
491	IL	Warren County	0 998824	0 914245	0 006181	724	4454	102478	0
492	IL	Washington County	0 997340	0 824784	0 200873	608	3820	128379	0
493	IL	Wayne County	0 997678	0 729311	0 020455	421	2206	102912	0
494	IL	White County	0 986484	0 741802	0 000000	493	2639	113729	0
495	IL	Whiteside County	0 982438	0 911078	0 027928	890	4580	110633	0
496	IL	Will County	0 985654	0 606940	0 026391	695	7144	127248	0
497	IL	Williamson County	0 954473	0 330028	0 004331	293	3234	100325	0
498	IL	Winnebago County	0 989427	0 618589	0 126839	728	4283	124464	0
499	IL	Woodford County	0 972844	0 875410	0 020122	819	5503	116018	0
500	MI	Alcona County	0 376681	0 098624	0 282110	253	2122	88614	0
501	MI	Alger County	0 181975	0 027403	0 472829	246	1749	75846	0
502	MI	Allegan County	0 451341	0 465240	0 204416	1545	4342	109096	0
503	MI	Alpena County	0 338732	0 210883	0 440294	330	1939	99505	0
504	MI	Antrim County	0 792348	0 168771	0 144520	626	2916	97724	0
505	MI	Arenac County	0 538920	0 351017	0 240191	589	2809	92664	0
506	MI	Baraga County	0 845707	0 024373	0 630952	177	1535	86460	0
507	MI	Barry County	0 963997	0 464600	0 372634	573	3014	111602	0
508	MI	Bay County	0 704087	0 636770	0 054178	832	4103	111751	0
509	MI	Benzie County	0 373772	0 096491	0 022548	569	3259	90774	1
510	MI	Berrien County	0 361078	0 456643	0 040881	1010	4519	109943	0
511	MI	Branch County	0 976707	0 701051	0 114179	719	3176	90781	0
512	MI	Calhoun County	0 986627	0 539862	0 180081	533	2798	107992	0
513	MI	Cass County	0 967997	0 591782	0 038115	966	3088	105791	0
514	MI	Charlevoix County	0 299705	0 153821	0 385775	295	2296	107457	0
515	MI	Cheboygan County	0 808348	0 089241	0 578444	278	1883	89115	0
516	MI	Chippewa County	0 578605	0 092893	0 413945	163	1636	73584	0
517	MI	Clare County	0 985414	0 176642	0 495634	411	1661	83100	0
518	MI	Clinton County	0 994577	0 700576	0 405672	698	3073	108053	0
519	MI	Crawford County	0 990699	0 003925	0 000000	148	3685	69791	10
520	MI	Delta County	0 587519	0 097176	0 450054	284	1527	97879	0
521	MI	Dickinson County	0 986129	0 057929	0 246551	347	1993	126015	0
522	MI	Eaton County	0 995649	0 633947	0 149117	499	3232	120319	0
523	MI	Emmet County	0 530362	0 134779	0 403053	297	3103	126354	0
524	MI	Genesee County	0 985056	0 334848	0 147134	557	4073	113986	0
525	MI	Gladwin County	0 981375	0 189700	0 343258	286	2325	86006	0
526	MI	Gogebic County	0 746296	0 008458	0 248792	172	2219	87957	1
527	MI	Grand Traverse County	0 773649	0 224380	0 091599	551	4606	113844	0
528	MI	Gratiot County	0 997408	0 760200	0 201292	700	3059	98882	0
529	MI	Hillsdale County	0 986293	0 604162	0 271225	598	2864	100311	0
530	MI	Houghton County	0 673765	0 045035	0 499209	214	1679	88194	0
531	MI	Huron County	0 391430	0 819765	0 172370	1015	3571	120597	0
532	MI	Ingham County	0 996879	0 541165	0 301432	583	3559	118247	0
533	MI	Ionia County	0 987905	0 694498	0 314953	735	2890	86541	0
534	MI	Iosco County	0 290445	0 134606	0 292981	500	2063	89372	0
535	MI	Iron County	0 963117	0 039867	0 196507	152	1468	98990	0
536	MI	Isabella County	0 993978	0 543409	0 422175	608	2595	97263	0
537	MI	Jackson County	0 976279	0 465751	0 224213	524	2871	106678	0
538	MI	Kalamazoo County	0 968411	0 429577	0 053540	1402	4862	123868	0
539	MI	Kalkaska County	0 982924	0 044772	0 149548	476	2197	73570	0
540	MI	Kent County	0 981682	0 348009	0 200170	1373	5566	126483	0
541	MI	Keweenaw County	0 089300	0 000762	0 000000	139	2269	92698	10
542	MI	Lake County	0 987717	0 049682	0 443380	157	2082	73631	0
543	MI	Lapeer County	0 986741	0 463156	0 290668	613	4266	116458	0
544	MI	Leelanau County	0 137562	0 291312	0 051639	952	4731	119581	0
545	MI	Lenawee County	0 985809	0 700033	0 129673	654	3582	110898	0
546	MI	Livingston County	0 970861	0 326474	0 291171	607	4395	143273	0
547	MI	Luce County	0 472341	0 016248	0 000000	263	1638	124586	0
548	MI	Mackinac County	0 486340	0 033920	0 692455	349	1789	93064	0
549	MI	Macomb County	0 843107	0 228664	0 069076	1203	8068	141722	0
550	MI	Manistee County	0 424597	0 137985	0 038590	400	2831	96234	0
551	MI	Marquette County	0 531475	0 019980	0 494550	185	1496	92414	0
552	MI	Mason County	0 398710	0 231719	0 255191	686	3058	97859	0
553	MI	Mecosta County	0 973123	0 340600	0 464418	450	2304	82111	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
554	MI	Menominee County	0 779935	0 164706	0 773368	376	1915	95272	0
555	MI	Midland County	0 987314	0 267322	0 050966	564	3480	133649	0
556	MI	Missaukee County	0 987668	0 243487	0 741791	814	2312	78285	0
557	MI	Monroe County	0 810374	0 615478	0 026987	755	4977	117027	0
558	MI	Montcalm County	0 982062	0 494356	0 239375	716	2803	92359	0
559	MI	Montmorency County	0 973640	0 062930	0 614665	287	2293	79991	0
560	MI	Muskegon County	0 348936	0 226032	0 209421	995	3768	100020	0
561	MI	Newaygo County	0 977924	0 213924	0 384147	845	3114	93301	0
562	MI	Oakland County	0 961058	0 086362	0 031629	1281	7446	182091	0
563	MI	Oceana County	0 413586	0 373166	0 135447	843	2951	98591	0
564	MI	Ogemaw County	0 982057	0 208601	0 674765	516	2415	74194	0
565	MI	Ontonagon County	0 350574	0 039288	0 479556	148	1148	96518	0
566	MI	Osceola County	0 987729	0 300087	0 704769	389	2206	80547	0
567	MI	Oscoda County	0 988516	0 038939	0 667914	328	2057	78217	0
568	MI	Otsego County	0 978318	0 110134	0 256572	194	2280	101171	0
569	MI	Ottawa County	0 346614	0 486952	0 121231	3264	6321	124356	0
570	MI	Presque Isle County	0 256509	0 189191	0 350748	337	2387	91594	0
571	MI	Roscommon County	0 899204	0 011345	0 000000	155	2717	90131	0
572	MI	Saginaw County	0 991606	0 614449	0 099446	579	3527	114494	0
573	MI	St. Clair County	0 870166	0 391607	0 156607	496	3884	118396	0
574	MI	St. Joseph County	0 966550	0 728369	0 084068	701	3543	100806	0
575	MI	Sanilac County	0 605954	0 720423	0 415141	650	2783	104538	0
576	MI	Schoolcraft County	0 625429	0 018445	0 067674	270	1451	84157	1
577	MI	Shiawassee County	0 996419	0 686657	0 264353	440	3245	106766	0
578	MI	Tuscola County	0 889218	0 623217	0 124181	742	3930	98787	0
579	MI	Van Buren County	0 560412	0 528792	0 032626	1015	3798	92773	0
580	MI	Washtenaw County	0 982676	0 415800	0 249292	688	5664	154307	0
581	MI	Wayne County	0 913554	0 057214	0 000000	2359	9530	118267	0
582	MI	Wexford County	0 982047	0 086827	0 501599	320	2398	90165	0
583	WI	Adams County	0 940777	0 287885	0 118357	824	3233	74715	0
584	WI	Ashland County	0 455041	0 076647	0 751264	239	1308	92278	0
585	WI	Barron County	0 969448	0 635341	0 538489	970	2278	102038	0
586	WI	Bayfield County	0 723109	0 103207	0 705791	295	1474	87591	0
587	WI	Brown County	0 859040	0 606910	0 647611	1311	3904	125236	0
588	WI	Buffalo County	0 964613	0 738388	0 634108	622	1795	111385	0
589	WI	Burnett County	0 933112	0 159933	0 656820	424	1603	84455	0
590	WI	Calumet County	0 805594	0 792349	0 697330	1176	3731	112191	0
591	WI	Chippewa County	0 970337	0 598199	0 733361	736	2283	104578	0
592	WI	Clark County	0.997174	0 548660	0 784958	813	2418	90673	0
593	WI	Columbia County	0 972461	0 660613	0 358539	790	3396	115862	0
594	WI	Crawford County	0 955724	0 679166	0 560449	482	1892	89196	0
595	WI	Dane County	0 970727	0 700009	0 410577	1241	4276	136053	0
596	WI	Dodge County	0.972794	0 733513	0 543775	1129	4001	104612	0
597	WI	Door County	0 203696	0 420933	0 634213	737	3374	116899	0
598	WI	Douglas County	0.884614	0 084193	0 468282	189	1313	100582	0
599	WI	Dunn County	0 986241	0 672241	0 570996	719	2363	90544	0
600	WI	Eau Claire County	0 988153	0 465335	0 593274	689	2533	105459	0
601	WI	Florence County	0 981041	0 067253	0 591308	281	1729	83486	0
602	WI	Fond du Lac County	0 943989	0 759952	0 615353	1062	3529	117732	0
603	WI	Forest County	0 969092	0 040761	0 275388	409	1621	71362	1
604	WI	Grant County	0 969971	0 845227	0 509891	814	2713	104524	0
605	WI	Green County	0 999015	0 784221	0 626323	996	3189	122893	0
606	WI	Green Lake County	0 931253	0 719476	0 476705	848	3380	104863	0
607	WI	Iowa County	0 992955	0 741396	0 530793	763	2847	101354	0
608	WI	Iron County	0 823767	0 021176	0 255343	679	1594	82335	11
609	WI	Jackson County	0 987214	0 345239	0 449766	717	2648	92251	0
610	WI	Jefferson County	0 955842	0 652391	0 347568	1129	3714	114291	0
611	WI	Juneau County	0 954580	0 397470	0 406154	705	2622	94493	0
612	WI	Kenosha County	0 361659	0 531261	0 303922	816	5022	117434	0
613	WI	Kewaunee County	0 315925	0 776237	0 706847	1157	3853	106874	0
614	WI	La Crosse County	0 943372	0 629238	0 585397	651	3109	113864	0
615	WI	Lafayette County	0 998402	0 879511	0 502392	940	3126	102885	0
616	WI	Langlade County	0 982881	0 215524	0 373029	879	2691	91560	0
617	WI	Lincoln County	0 973617	0 151288	0 665757	487	2110	97886	0
618	WI	Manitowoc County	0 395999	0 657321	0 698904	1228	3702	108818	0
619	WI	Marathon County	0 980226	0 535942	0 576136	1030	2757	110437	0
620	WI	Marinette County	0 904344	0 162695	0 617953	683	2623	97311	0
621	WI	Marquette County	0 980788	0 464902	0 430073	512	2078	99146	0
622	WI	Menominee County	0 980747	0 000956	0 000000	578	1358	84245	25
623	WI	Milwaukee County	0 203028	0 056682	0 008057	1918	7521	124837	1

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
624	WI	Monroe County	0 991697	0 600763	0 608095	688	2870	87659	0
625	WI	Oconto County	0 868526	0 327012	0 713432	738	2520	92556	0
626	WI	Oneida County	0 909893	0 044145	0 028190	605	2231	105913	1
627	WI	Outagamie County	0 993691	0 642949	0 629770	1221	3625	116377	0
628	WI	Ozaukee County	0 207796	0 530607	0 558680	975	4458	169716	0
629	WI	Pepin County	0 934136	0 763691	0 658518	610	2196	99566	0
630	WI	Pierce County	0 974479	0 739559	0 566909	659	2796	115293	0
631	WI	Polk County	0 959210	0 481026	0 665064	571	2440	96349	0
632	WI	Portage County	0 980004	0 514907	0 306200	847	3338	104294	0
633	WI	Price County	0 979762	0 117991	0 666608	299	1684	100230	0
634	WI	Racine County	0 420631	0 624739	0 176466	1059	4382	124803	0
635	WI	Richland County	0 994597	0 722077	0 577614	578	2228	86961	0
636	WI	Rock County	0 992128	0 744044	0 298580	832	3803	113397	0
637	WI	Rusk County	0 980909	0 287975	0 734812	494	1581	83331	0
638	WI	St Croix County	0 981121	0 667584	0 592979	750	2840	127594	0
639	WI	Sauk County	0 987274	0 625803	0 516358	837	2947	111690	0
640	WI	Sawyer County	0 930458	0 058912	0 588121	508	1963	82565	0
641	WI	Shawano County	0 981501	0 521640	0.651669	1033	3199	84245	23
642	WI	Sheboygan County	0 404120	0 630056	0 630239	1071	3691	122331	0
643	WI	Taylor County	0 990228	0 370892	0 709633	627	1825	96173	0
644	WI	Trempealeau County	0 989350	0 741935	0.501632	751	2282	104626	0
645	WI	Vernon County	0 973609	0 718421	0 646133	654	2499	89643	0
646	WI	Vilas County	0 857436	0 014508	0.000000	1096	5486	94764	12
647	WI	Walworth County	0 963316	0 636070	0 291054	1007	4537	115930	0
648	WI	Washburn County	0 949150	0 166123	0 538450	376	1532	87557	0
649	WI	Washington County	0 988369	0 533855	0 568445	1074	5237	132260	0
650	WI	Waukesha County	0 957081	0 321114	0 323690	952	6534	155811	0
651	WI	Waupaca County	0 981398	0 502940	0 683982	819	3048	105567	0
652	WI	Waushara County	0 982180	0 417253	0 233678	948	3536	104829	0
653	WI	Winnebago County	0 757888	0 605137	0 597764	851	3511	119513	0
654	WI	Wood County	0 979427	0 436233	0 528971	921	3421	122683	0
655	MN	Aitkin County	0 911790	0 144343	0 279633	232	1206	86507	0
656	MN	Anoka County	0 950055	0 227864	0 061307	827	6451	120421	0
657	MN	Becker County	0 906811	0 450322	0 236521	494	1596	95617	0
658	MN	Beltrami County	0 819905	0 140275	0 268278	187	1111	86365	0
659	MN	Benton County	0 988562	0 703229	0 307408	1048	2542	95461	0
660	MN	Big Stone County	0 941416	0 824348	0 076928	421	2034	99464	0
661	MN	Blue Earth County	0 982321	0 796158	0 028631	813	4101	101307	0
662	MN	Brown County	0 987486	0 888620	0 129195	864	3746	112049	0
663	MN	Carlton County	0 982969	0 205981	0 540515	206	1722	95963	0
664	MN	Carver County	0 949172	0 726154	0 473180	958	5190	138763	0
665	MN	Cass County	0 835723	0 155033	0 322334	206	1055	81461	0
666	MN	Chippewa County	0 991451	0 876138	0 025146	623	3116	109204	0
667	MN	Chisago County	0 943838	0 518473	0 324198	423	3198	108649	0
668	MN	Clay County	0 992882	0 847526	0 062607	525	2495	100955	0
669	MN	Clearwater County	0 965937	0 331260	0 134254	296	1094	78298	0
670	MN	Cook County	0 434446	0 001345	0 000000	87	2267	113817	0
671	MN	Cottonwood County	0 986240	0 915311	0 032518	838	3464	109482	0
672	MN	Crow Wing County	0 861728	0 204869	0 358056	265	1462	101408	0
673	MN	Dakota County	0 971613	0 606642	0 127995	897	4500	143070	0
674	MN	Dodge County	0 999722	0 857286	0 204112	759	3517	106881	0
675	MN	Douglas County	0 881040	0 640757	0 439861	475	2080	96356	0
676	MN	Faribault County	0 988936	0 907978	0 034202	715	3749	109326	0
677	MN	Fillmore County	0 998957	0 804562	0 260043	650	2668	103014	0
678	MN	Freeborn County	0 979257	0 809293	0 060712	763	3537	103982	0
679	MN	Goodhue County	0 971963	0 781886	0 393166	801	3570	113512	0
680	MN	Grant County	0 950010	0 769583	0 061248	478	2272	112598	0
681	MN	Hennepin County	0 917888	0 222269	0 214677	1184	8076	160559	0
682	MN	Houston County	0 981543	0 761178	0 289458	730	2531	102573	0
683	MN	Hubbard County	0 923077	0 190387	0 090525	338	1584	84292	0
684	MN	Isanti County	0 971654	0 468169	0 199593	369	3375	101429	0
685	MN	Itasca County	0 910304	0 063202	0 162397	145	1484	90754	0
686	MN	Jackson County	0 975596	0 892694	0 018139	782	4002	113431	0
687	MN	Kanabec County	0 984149	0 433208	0 404207	320	1893	87720	0
688	MN	Kandiyohi County	0 923669	0 707455	0 092776	1276	2955	111108	0
689	MN	Kittson County	0 994169	0 687878	0 008604	315	1735	128549	0
690	MN	Koochiching County	0 983526	0 034640	0 167034	126	950	95942	1
691	MN	Lac qui Parle County	0 983050	0 827370	0 034986	452	2403	106150	0
692	MN	Lake County	0 701872	0 003916	0 000000	54	2262	86494	0
693	MN	Lake of the Woods County	0 730503	0 124914	0 081647	133	950	90287	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
694	MN	Le Sueur County	0 946487	0 714235	0 141268	788	3632	109983	0
695	MN	Lincoln County	0 979197	0 743207	0 109379	491	1698	108656	0
696	MN	Lyon County	0 989960	0 864158	0 041098	690	2870	112245	0
697	MN	McLeod County	0.972791	0 795665	0 309049	732	3529	114305	0
698	MN	Mahnomen County	0 953949	0 524154	0 192941	285	1592	88695	0
699	MN	Marshall County	0 977632	0 656562	0 029137	333	1666	110816	0
700	MN	Martin County	0 972297	0 908935	0 021702	845	4360	119127	0
701	MN	Meeker County	0 943353	0 773103	0 159996	819	2857	107613	0
702	MN	Mille Lacs County	0.842618	0 387383	0 513037	486	2232	102052	0
703	MN	Morrison County	0 975012	0 587619	0 406929	714	2090	87144	0
704	MN	Mower County	0 999705	0 862167	0 098472	718	3353	114345	0
705	MN	Murray County	0 979031	0 833130	0 059715	647	2786	106854	0
706	MN	Nicollet County	0 968621	0 835722	0 110103	1116	3847	98570	0
707	MN	Nobles County	0 990438	0 909759	0 054392	741	3257	115849	0
708	MN	Norman County	0 999373	0 816056	0 044596	421	2093	121112	0
709	MN	Olmsted County	0 997719	0 731749	0 285655	763	3468	138952	0
710	MN	Otter Tail County	0 889803	0 647994	0 364077	514	1763	98672	0
711	MN	Pennington County	0 997084	0 709813	0 083563	259	1225	106698	0
712	MN	Pine County	0 983605	0 291508	0 458014	384	1608	78379	0
713	MN	Pipestone County	0 999444	0 847331	0 137716	574	2415	107166	0
714	MN	Polk County	0 986288	0 826926	0 039682	466	2359	111013	0
715	MN	Pope County	0 934285	0 723061	0 273997	464	1975	93490	0
716	MN	Ramsey County	0.915724	0 021479	0 010931	6179	15256	137516	1
717	MN	Red Lake County	0 999812	0 661966	0 166458	324	1280	101876	0
718	MN	Redwood County	0 998402	0 873219	0 038427	773	3597	104260	0
719	MN	Renville County	0 995644	0 953920	0 036486	777	3841	117224	0
720	MN	Rice County	0 964050	0 714450	0 228944	960	3910	106793	0
721	MN	Rock County	0 999524	0 875190	0 058657	827	3380	108188	0
722	MN	Roseau County	0 990579	0 503990	0 128189	249	1330	111649	0
723	MN	St Louis County	0 907461	0 038447	0 269956	215	1542	104436	0
724	MN	Scott County	0 968091	0 576959	0 363559	835	6070	129097	0
725	MN	Sherburne County	0 967987	0 421259	0 094866	840	4015	92088	0
726	MN	Sibley County	0 980439	0 827765	0 138753	1047	3774	106461	0
727	MN	Stearns County	0 967336	0 748096	0 439537	1032	2724	99952	0
728	MN	Steele County	0 993947	0 842459	0 162386	790	3729	119154	0
729	MN	Stevens County	0 977103	0 795943	0 021373	681	2425	103447	0
730	MN	Swift County	0 988275	0 819310	0 057858	533	2433	106258	0
731	MN	Todd County	0 961884	0 655263	0 507057	598	1821	85749	0
732	MN	Traverse County	0 979742	0 844173	0 009468	480	2421	138661	0
733	MN	Wabasha County	0 954886	0 731145	0 473087	712	2899	100182	0
734	MN	Wadena County	0 986070	0 500194	0 303896	548	1505	83764	0
735	MN	Waseca County	0 977916	0 875776	0 104726	708	3803	112266	0
736	MN	Washington County	0 925667	0 401951	0 094731	1181	6860	140083	0
737	MN	Watonwan County	0 987716	0 898001	0 021110	746	3409	106915	0
738	MN	Wilkin County	0 999740	0 874905	0 018943	451	2348	114603	0
739	MN	Winona County	0 976173	0 725008	0 540247	787	3057	104456	0
740	MN	Wright County	0 924987	0 644404	0 348824	735	4035	114054	0
741	MN	Yellow Medicine County	0 992885	0 840947	0 042125	568	2790	108168	0
742	IA	Adair County	0 998243	0 902845	0 016359	565	2321	107897	0
743	IA	Adams County	0 995613	0 884621	0 005438	476	1892	112814	0
744	IA	Allamakee County	0 970936	0 785949	0 328593	687	2495	103515	0
745	IA	Appanoose County	0 961066	0 751233	0 017301	284	1795	100697	0
746	IA	Audubon County	0 998999	0 946715	0 002453	863	3471	105926	0
747	IA	Benton County	0 997132	0 931703	0 024856	878	3959	115896	0
748	IA	Black Hawk County	0 991983	0 824804	0 030499	929	4466	112652	0
749	IA	Boone County	0 996419	0 902469	0 004196	771	4435	108046	0
750	IA	Bremer County	0 996128	0 844404	0 164681	931	4476	112889	0
751	IA	Buchanan County	0 996348	0 911084	0 058752	865	3963	105181	0
752	IA	Buena Vista County	0 990896	0 929428	0 004171	1137	4480	109015	0
753	IA	Butler County	0 997970	0 849191	0 044717	895	3720	105777	0
754	IA	Calhoun County	0 996287	0 946944	0 000882	803	4714	107294	0
755	IA	Carroll County	0 998390	0 987299	0 002872	1346	4153	114223	0
756	IA	Cass County	0 998797	0 961707	0 005032	686	2891	118158	0
757	IA	Cedar County	0 995815	0 913391	0 012896	859	3953	120068	0
758	IA	Cerro Gordo County	0 988259	0 848001	0 007197	731	4108	117501	0
759	IA	Cherokee County	0 999685	0 910249	0 016606	940	3980	106143	0
760	IA	Chickasaw County	0 998363	0 851072	0 081408	900	3547	111934	0
761	IA	Clarke County	0 998609	0 856746	0 016323	291	1678	106143	0
762	IA	Clay County	0 993667	0 864587	0 001256	825	4125	114643	0
763	IA	Clayton County	0 982337	0.916724	0 283145	842	3283	102729	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
764	IA	Clinton County	0 978581	0 827543	0 034845	909	3692	110708	0
765	IA	Crawford County	0 999119	0 907880	0 011585	740	3101	98645	0
766	IA	Dallas County	0 991104	0 831655	0 003446	691	4398	120008	0
767	IA	Davis County	0 996735	0 854756	0 062016	359	1677	94385	0
768	IA	Decatur County	0 997920	0 767653	0 012533	388	1402	83520	0
769	IA	Delaware County	0 997936	0 908840	0 199925	1315	4248	106292	0
770	IA	Des Moines County	0 968244	0 722602	0 026864	661	4120	115822	0
771	IA	Dickinson County	0 943924	0 829282	0 010817	875	3833	127438	0
772	IA	Dubuque County	0 986365	0 883394	0 311392	1329	4061	110559	0
773	IA	Emmet County	0 983513	0 887554	0 001697	713	4115	98821	1
774	IA	Fayette County	0 999326	0 858489	0 176250	933	3362	101043	0
775	IA	Floyd County	0 998530	0 897635	0 007781	793	3482	111243	0
776	IA	Franklin County	0 999017	0 921066	0 010472	850	3876	109746	0
777	IA	Fremont County	0 989420	0 923935	0 003414	675	2673	118192	0
778	IA	Greene County	0 995236	1 008589	0 000350	705	3847	115720	13
779	IA	Grundy County	0 999936	0 986230	0 009951	999	4707	121816	0
780	IA	Guthrie County	0 995857	0 870056	0 007203	633	3089	106603	0
781	IA	Hamilton County	0 998677	0 900520	0 004620	1059	4927	132592	0
782	IA	Hancock County	0 996451	0 900468	0 007410	816	4279	109089	0
783	IA	Hardin County	0 998813	0 912171	0 001753	1069	4060	118599	0
784	IA	Harrison County	0 994104	0 894992	0 001232	623	2726	103176	0
785	IA	Henry County	0 994864	0 812175	0 009874	711	3801	119811	0
786	IA	Howard County	0 999244	0 860664	0 109331	700	3136	114061	0
787	IA	Humboldt County	0 996979	1 009926	0 006695	803	4435	118382	9
788	IA	Ida County	0 998801	0 987401	0 007997	801	3146	105594	0
789	IA	Iowa County	0 998486	0 855933	0 017511	735	2884	117915	0
790	IA	Jackson County	0 978975	0 851236	0 112330	739	2772	107308	0
791	IA	Jasper County	0.995991	0.922868	0.020033	776	3364	123211	0
792	IA	Jefferson County	0 996850	0 814915	0 016176	576	2730	108276	0
793	IA	Johnson County	0 985773	0 723453	0 049442	864	4451	122304	0
794	IA	Jones County	0 997600	0 874325	0 052290	985	3569	99918	0
795	IA	Keokuk County	0 998707	0 869681	0 004075	638	2864	110383	0
796	IA	Kossuth County	0 998579	0 987584	0 015028	764	4547	109604	0
797	IA	Lee County	0 960314	0 803517	0 045955	683	2999	107186	0
798	IA	Linn County	0 990195	0 760552	0 037957	724	4444	129693	0
799	IA	Louisa County	0 962330	0 743622	0 001590	596	3394	112706	1
800	IA	Lucas County	0 991666	0 796030	0 024712	286	1653	108358	0
801	IA	Lyon County	0 999772	0 924385	0 035682	1233	3802	106170	0
802	IA	Madison County	0 997937	0 851139	0 002642	561	2580	113729	1
803	IA	Mahaska County	0.995638	0.861745	0.023850	910	3907	105520	0
804	IA	Marion County	0 971396	0 756979	0 033093	600	2992	118511	0
805	IA	Marshall County	0 998780	0 854067	0 014080	802	3788	122642	0
806	IA	Mills County	0 992873	0 851325	0 002191	673	3314	109949	0
807	IA	Mitchell County	0 998992	0 876446	0 049672	1002	3717	124898	0
808	IA	Monona County	0 991798	0 885518	0 002008	626	2634	103914	0
809	IA	Monroe County	0 998389	0 806187	0 044018	328	1630	107294	0
810	IA	Montgomery County	0 997809	0 885000	0 006614	652	2561	113905	0
811	IA	Muscatine County	0 976840	0 782982	0 039235	735	3929	126997	0
812	IA	O'Brien County	0 999701	0 987197	0 009215	998	4496	106942	0
813	IA	Osceola County	0 998193	1 021741	0 032591	897	4451	110613	9
814	IA	Page County	0 999021	0 931255	0 005491	634	2435	110735	0
815	IA	Palo Alto County	0 990217	0 938619	0 004677	786	3840	112571	0
816	IA	Plymouth County	0 999554	0 937623	0 014576	1055	3681	113038	0
817	IA	Pocahontas County	0 997539	0 972106	0 003862	757	4359	107911	0
818	IA	Polk County	0 962062	0 630556	0 009397	664	4096	136405	0
819	IA	Pottawattamie County	0 994086	0 888804	0 003748	820	3385	112740	0
820	IA	Poweshiek County	0 998046	0 910621	0 014505	626	2995	119730	0
821	IA	Ringgold County	0 997683	0 852185	0 001917	323	1458	96295	1
822	IA	Sac County	0 995530	0 988159	0 007685	993	4165	107897	0
823	IA	Scott County	0 978023	0 795786	0 047835	965	5650	130025	0
824	IA	Shelby County	0 999179	0 934909	0 004707	867	3473	110004	0
825	IA	Sioux County	0 999090	1 008712	0 041501	2104	4825	107166	9
826	IA	Story County	0 998540	0 903348	0 019025	792	4348	107436	0
827	IA	Tama County	0 998519	0 870450	0 017513	674	3894	108703	0
828	IA	Taylor County	0 998369	0 816166	0 014307	414	1910	98333	0
829	IA	Union County	0 996322	0 869825	0 005260	507	2153	105560	0
830	IA	Van Buren County	0 989125	0 777372	0 025127	375	1951	93707	0
831	IA	Wapello County	0 990431	0 705623	0 017685	484	2765	102567	0
832	IA	Warren County	0 997274	0 826748	0 039354	466	3059	114548	0
833	IA	Washington County	0 996525	0 850183	0 015409	1029	3755	113309	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
834	IA	Wayne County	0 997199	0 840410	0 012621	296	1574	97907	0
835	IA	Webster County	0 996070	0 892276	0 004648	768	4401	111378	0
836	IA	Winnebago County	0 997248	0 905067	0 019499	645	3761	122283	0
837	IA	Winneshiek County	0 999609	0 810372	0 299449	864	2821	107755	0
838	IA	Woodbury County	0 994554	0 791848	0 002080	843	3039	113817	0
839	IA	Worth County	0 995729	0 877434	0 008399	723	3514	125175	0
840	IA	Wright County	0 996882	0 951560	0 002304	694	4566	125914	0
841	MO	Adair County	0 997134	0 735025	0 030400	196	1339	91242	0
842	MO	Andrew County	0 996980	0 815542	0 087225	418	2488	100507	0
843	MO	Atchison County	0 995172	0 872062	0 002415	512	2410	112584	0
844	MO	Audrain County	0 995015	0 849715	0 019900	498	2472	102201	0
845	MO	Barry County	0 985041	0 586441	0 105550	829	2311	91208	0
846	MO	Barton County	0 995934	0 818018	0 060261	307	1965	99512	0
847	MO	Bates County	0 996592	0 792625	0 056804	388	1710	101754	0
848	MO	Benton County	0 937578	0 529930	0 106382	289	1612	76327	0
849	MO	Bollinger County	0 999270	0 497197	0 028189	215	1780	75562	0
850	MO	Boone County	0 991599	0 619897	0 008427	327	2883	116330	1
851	MO	Buchanan County	0 988329	0 691301	0 063070	390	2656	108317	0
852	MO	Butler County	0 997966	0 572283	0 000638	455	2814	86663	0
853	MO	Caldwell County	0 999123	0 844933	0 021723	280	1606	92488	0
854	MO	Callaway County	0 990358	0 631954	0 025124	281	2438	100507	0
855	MO	Camden County	0 924234	0 388903	0 185498	188	1475	97426	0
856	MO	Cape Girardeau County	0 986917	0 682856	0 143157	424	3101	111460	0
857	MO	Carroll County	0 988952	0 848076	0 010751	407	2280	93660	0
858	MO	Carter County	0 997171	0 168578	0 044845	122	1475	63993	1
859	MO	Cass County	0 994860	0 728141	0 055725	383	3338	115124	0
860	MO	Cedar County	0 954723	0 616689	0 067176	215	1805	82599	0
861	MO	Chariton County	0 983881	0 834250	0 009443	464	2268	98374	0
862	MO	Christian County	0 998423	0 585790	0 322899	331	3520	97609	0
863	MO	Clark County	0 991043	0 753960	0.009234	321	1917	82288	0
864	MO	Clay County	0 968957	0 513761	0 005428	423	3767	124593	1
865	MO	Clinton County	0 988879	0.774594	0 041408	400	2269	106705	0
866	MO	Cole County	0 981083	0 747164	0 113317	245	2448	110505	0
867	MO	Cooper County	0 990888	0 825996	0 022350	451	2077	92563	0
868	MO	Crawford County	0 998317	0 424349	0 033875	118	1675	94974	0
869	MO	Dade County	0 968478	0 805443	0 073819	272	1522	87801	0
870	MO	Dallas County	0 997636	0 657160	0 411141	340	1951	80472	0
871	MO	Daviess County	0 996284	0 768407	0 028121	287	1634	85180	0
872	MO	DeKalb County	0 996270	0 776552	0 067878	310	1837	75487	0
873	MO	Dent County	0 998710	0 454993	0 053991	126	1557	90659	0
874	MO	Douglas County	0 999900	0 577303	0 445827	245	1721	63404	0
875	MO	Dunklin County	0 997272	0 827024	0 000000	706	3044	77878	0
876	MO	Franklin County	0 991858	0 502064	0 127990	313	3221	110884	0
877	MO	Gasconade County	0 990374	0 592416	0 069316	184	1906	106603	0
878	MO	Gentry County	0 999418	0 781410	0 014098	312	1562	94324	0
879	MO	Greene County	0 995829	0 660855	0 308635	361	3887	112638	0
880	MO	Grundy County	0 995065	0 811399	0 028646	290	1759	96687	0
881	MO	Harrison County	0 998197	0 860100	0 011792	233	1473	95915	0
882	MO	Henry County	0 958890	0 714374	0 050032	284	1963	93341	0
883	MO	Hickory County	0 968159	0 683201	0 203416	224	1541	70651	0
884	MO	Holt County	0 984683	0 786801	0 002607	494	2684	96742	1
885	MO	Howard County	0 989747	0 801478	0 011960	289	2108	91465	0
886	MO	Howell County	0 999392	0 626955	0 258800	263	1796	83249	0
887	MO	Iron County	0 998791	0 194371	0 000000	188	1733	76544	0
888	MO	Jackson County	0 981595	0 346685	0 024381	479	4970	126056	0
889	MO	Jasper County	0 997578	0 687031	0 101724	529	2439	99823	0
890	MO	Jefferson County	0 989017	0 284499	0 298079	222	3797	107667	0
891	MO	Johnson County	0 997045	0 697930	0 033823	383	2364	82708	0
892	MO	Knox County	0 997894	0 829389	0 038162	276	1656	91249	0
893	MO	Laclede County	0 997211	0 621338	0 477062	282	1975	90774	0
894	MO	Lafayette County	0 985080	0 884235	0 038966	579	2771	109645	0
895	MO	Lawrence County	0 999506	0 848400	0 212837	571	2796	87787	0
896	MO	Lewis County	0 988597	0 770485	0 016198	350	1845	87144	0
897	MO	Lincoln County	0 984497	0 627667	0 049467	509	3249	102722	0
898	MO	Linn County	0 998211	0 851010	0 054104	300	1512	95712	0
899	MO	Livingston County	0 992805	0 790767	0 021177	297	2097	103190	0
900	MO	McDonald County	0 999644	0 577141	0 033075	1013	2476	77404	0
901	MO	Macon County	0 989214	0 742426	0 017445	211	1456	100704	0
902	MO	Madison County	0 998251	0 350861	0 008255	163	1693	86907	1
903	MO	Maries County	0 995822	0 690684	0 113155	181	1311	85573	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
904	MO	Marion County	0 986553	0 784267	0 059833	375	2259	101225	0
905	MO	Mercer County	0 998347	0 722882	0 007618	296	1576	84855	0
906	MO	Miller County	0 987151	0 638474	0 012265	541	1855	89867	0
907	MO	Mississippi County	0 963279	1 003056	0 000801	705	3592	81949	13
908	MO	Moniteau County	0 994438	0 814424	0 047309	428	2008	105262	0
909	MO	Monroe County	0 963838	0 740529	0 027497	362	2112	107294	0
910	MO	Montgomery County	0 993994	0 651774	0 003409	391	2701	100826	1
911	MO	Morgan County	0 973191	0 527510	0 072972	566	1816	86379	0
912	MO	New Madrid County	0 971339	0 849960	0 000000	641	3265	80242	0
913	MO	Newton County	0 999611	0 638553	0 086552	849	2798	87787	0
914	MO	Nodaway County	0 998714	0 905181	0 014332	399	1972	91878	0
915	MO	Oregon County	0 999768	0 497640	0 118160	256	1514	68098	0
916	MO	Osage County	0 988104	0 816658	0 044117	417	1534	107680	0
917	MO	Ozark County	0 989024	0 524030	0 338215	257	1555	81129	0
918	MO	Pemiscot County	0 962278	0 924759	0 000000	637	3012	81861	0
919	MO	Perry County	0 980166	0 689426	0 110738	394	2542	103264	0
920	MO	Pettis County	0 998026	0 819841	0 028568	385	2124	107396	0
921	MO	Phelps County	0 997883	0 466191	0 023793	123	1840	94161	0
922	MO	Pike County	0 982608	0 751068	0 006241	432	2371	94784	0
923	MO	Platte County	0 983873	0 700999	0 004576	347	2970	129856	1
924	MO	Polk County	0 991811	0 847575	0 324205	376	2116	89372	0
925	MO	Pulaski County	0 992042	0 396975	0 119023	152	1560	74146	0
926	MO	Putnam County	0 996646	0 767722	0 011206	267	1158	86541	0
927	MO	Ralls County	0 973515	0 759435	0 011220	350	2289	100968	0
928	MO	Randolph County	0 989204	0 716186	0 013737	254	1702	91973	0
929	MO	Ray County	0 992837	0 760858	0 007199	377	2544	103312	0
930	MO	Reynolds County	0 996055	0 172732	0 021025	96	1823	74790	0
931	MO	Ripley County	0 996521	0 378589	0 013322	147	1551	64792	1
932	MO	St Charles County	0.945923	0 568210	0 057459	531	5125	127018	0
933	MO	St Clair County	0 964061	0 593892	0 048662	201	1412	80093	0
934	MO	Ste Genevieve County	0 987142	0 524294	0 022366	268	2113	97426	0
935	MO	St Francois County	0 993515	0 406391	0 042593	358	2844	89169	0
936	MO	St Louis County	0 969727	0 166425	0 000000	752	5869	164040	0
937	MO	Saline County	0 988179	0 8556944	0 001390	520	2505	103400	1
938	MO	Schuylerville County	0 999065	0 838477	0 058161	211	1172	89576	0
939	MO	Scotland County	0.998210	0 772143	0 096079	334	1823	90415	0
940	MO	Scott County	0 988218	0 812867	0 012812	584	3055	93836	0
941	MO	Shannon County	0 999847	0 186828	0 071799	111	1191	59015	0
942	MO	Shelby County	0 996932	0 852800	0 001302	420	1914	102275	1
943	MO	Stoddard County	0 997832	0 827612	0 001356	661	3464	88648	0
944	MO	Stone County	0 906691	0 464587	0 446066	317	2355	89982	0
945	MO	Sullivan County	0 999265	0 792044	0 013984	193	1160	89156	0
946	MO	Taney County	0 970657	0 396734	0 117372	115	1536	101253	0
947	MO	Texas County	0 999412	0 609378	0 445284	224	1652	75867	0
948	MO	Vernon County	0 996311	0 753505	0 069337	254	1872	98902	0
949	MO	Warren County	0 986153	0 457762	0 005391	439	4550	109658	1
950	MO	Washington County	0 996519	0 230001	0 017565	305	1718	73320	1
951	MO	Wayne County	0 983139	0 191038	0 013536	119	1300	67102	1
952	MO	Webster County	0 999441	0 762889	0 423597	445	2259	80269	0
953	MO	Worth County	0 999126	0 785702	0 012621	226	1566	91309	0
954	MO	Wright County	0 998695	0 725019	0 619215	376	1819	71769	0
955	MO	St Louis city	0 936142	0 000000	0 000000	0	0	122683	0
956	ND	Adams County	0 999067	0 940357	0 012619	97	549	112469	0
957	ND	Barnes County	0 985740	0 898963	0 025023	263	1259	102858	0
958	ND	Benson County	0 964695	0 875082	0 021521	151	1024	80418	0
959	ND	Billings County	0 998287	1 111182	0 048065	31	427	116770	9
960	ND	Bottineau County	0.982829	0 889582	0 009700	147	1013	113810	0
961	ND	Bowman County	0 995761	0 911546	0 020248	71	504	113160	1
962	ND	Burke County	0 977216	0 791986	0 009021	105	765	110125	0
963	ND	Burleigh County	0 979051	0 839411	0 046116	106	741	113715	0
964	ND	Cass County	0 998728	0 947303	0 008512	371	2063	110234	0
965	ND	Cavalier County	0 986002	0 897636	0 002487	250	1347	131231	0
966	ND	Dickey County	0 990664	0 867206	0 036509	240	893	106217	0
967	ND	Divide County	0 973101	0 900678	0 009728	92	858	113851	1
968	ND	Dunn County	0 965244	1 051566	0 052667	68	568	87679	9
969	ND	Eddy County	0 981110	0 912498	0 037130	147	858	108595	0
970	ND	Emmons County	0 971163	0 863307	0 219787	145	724	91370	0
971	ND	Foster County	0 982239	0 900932	0 010715	212	1124	105330	0
972	ND	Golden Valley County	0 999626	0 788185	0 056375	80	525	130100	0
973	ND	Grand Forks County	0 998595	0 835889	0 005533	429	1991	93010	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
974	ND	Grant County	0 996125	0 959682	0 116877	94	586	83994	0
975	ND	Griggs County	0 989296	0 873609	0 024441	211	1053	100982	0
976	ND	Hettinger County	0 998720	0 950015	0 059190	138	769	109170	0
977	ND	Kidder County	0 943029	0 836728	0 086128	102	669	81400	0
978	ND	LaMoure County	0 996846	0 911225	0 081321	256	1165	101882	0
979	ND	Logan County	0 981773	0 942542	0 120436	156	774	90463	0
980	ND	McHenry County	0 980324	0 874286	0 040379	111	675	87903	0
981	ND	McIntosh County	0 980060	0 872771	0 092147	158	766	104118	0
982	ND	McKenzie County	0 958449	0 664218	0 006237	104	853	102025	0
983	ND	McLean County	0 906438	0 835393	0 039696	132	947	98184	0
984	ND	Mercer County	0 939720	0 794591	0 055201	103	787	112388	0
985	ND	Morton County	0 990250	1 000607	0 187885	124	644	92989	9
986	ND	Mounttrail County	0 939649	0 857211	0 020923	101	908	93253	0
987	ND	Nelson County	0 973118	0 879717	0 028526	216	1127	109184	0
988	ND	Oliver County	0 989508	0 829692	0 139547	116	725	90754	0
989	ND	Pembina County	0 997338	0 839132	0 000578	470	2378	129334	1
990	ND	Pierce County	0 940544	0 899888	0 052579	131	880	98042	0
991	ND	Ramsey County	0 911833	0 842623	0 010051	196	1141	112625	0
992	ND	Ransom County	0 998398	0 878347	0 037508	261	1177	98299	0
993	ND	Renville County	0 980576	0 899464	0 015624	159	1017	98611	0
994	ND	Richland County	0 993848	0 869495	0 019029	404	2185	95434	0
995	ND	Rolette County	0 960762	0 904624	0 051934	137	897	74112	0
996	ND	Sargent County	0 990386	0 901516	0 036878	272	1188	128698	0
997	ND	Sheridan County	0 966173	0 838234	0 079475	102	808	113295	0
998	ND	Sioux County	0 969708	1 065017	0 073349	46	564	63736	9
999	ND	Slope County	0 998931	1 007974	0 008615	59	478	101049	9
1000	ND	Stark County	0 998284	0 982783	0 152617	124	840	96085	0
1001	ND	Steele County	0 995626	0 964695	0 009749	279	1396	111263	0
1002	ND	Stutsman County	0 966558	0 892960	0 061470	193	956	107172	0
1003	ND	Towner County	0 984399	0 900876	0 002927	188	1026	124850	1
1004	ND	Trail County	0 999250	0 908298	0 000772	421	2381	105174	1
1005	ND	Walsh County	0 990598	0 898583	0 001429	462	1945	121653	0
1006	ND	Ward County	0 978911	0 901112	0 035167	154	1024	101171	0
1007	ND	Wells County	0 985013	0 922874	0 045496	180	979	118423	0
1008	ND	Williams County	0 963909	0 892468	0 004306	96	765	112476	1
1009	SD	Aurora County	0 993895	0 838296	0 028897	285	763	96538	0
1010	SD	Beadle County	0 995645	0 899194	0 055926	281	1084	109760	0
1011	SD	Bennett County	0 995549	1 038492	0 004195	68	476	110749	9
1012	SD	Bon Homme County	0 969176	0 894140	0 045044	558	1770	103156	0
1013	SD	Brookings County	0 987202	0 874038	0 074316	438	1748	92292	0
1014	SD	Brown County	0 989294	0 936276	0 027643	305	1220	115131	0
1015	SD	Brule County	0 967501	0 947745	0 031406	207	840	102059	0
1016	SD	Buffalo County	0 965478	0 926944	0 009761	145	644	58317	1
1017	SD	Butte County	0 992123	0 863856	0 146387	78	432	91282	0
1018	SD	Campbell County	0 953924	0 886940	0 086895	159	1017	116269	0
1019	SD	Charles Mix County	0 954800	0 978917	0 042257	298	1113	94642	0
1020	SD	Clark County	0 989805	0 872297	0 022527	333	968	121247	0
1021	SD	Clay County	0 987780	0 898113	0 005923	460	2344	88973	0
1022	SD	Codington County	0 959048	0 892699	0 205704	368	1454	103339	0
1023	SD	Corson County	0 977680	1 075332	0 029745	54	373	78799	9
1024	SD	Custer County	0 999070	0 463643	0 101622	54	501	108392	0
1025	SD	Davison County	0 996698	0 971205	0 080118	308	1418	111053	0
1026	SD	Day County	0 942589	0 852631	0 163696	211	992	119303	0
1027	SD	Deuel County	0 979316	0 854764	0 242241	314	1249	95299	0
1028	SD	Dewey County	0 941528	1 261474	0 063883	34	435	67861	9
1029	SD	Douglas County	0 998745	0 909568	0 111281	397	1297	78610	0
1030	SD	Edmunds County	0 995154	0 875472	0 062248	208	805	109177	0
1031	SD	Fall River County	0 994676	0 875407	0.002402	169	324	102296	1
1032	SD	Faulk County	0 994497	0 874887	0 008521	188	791	113885	0
1033	SD	Grant County	0 992113	0 855703	0 166253	384	1442	104199	0
1034	SD	Gregory County	0 964391	0 924334	0 114608	155	797	101015	0
1035	SD	Haakon County	0 992198	1 037982	0 001465	66	545	124512	9
1036	SD	Harlin County	0 950197	0 845879	0 155041	384	1556	90490	0
1037	SD	Hand County	0 997478	0 936527	0 049472	179	700	133209	0
1038	SD	Hanson County	0 997761	0 881271	0 069604	356	1626	84062	0
1039	SD	Harding County	0 997356	0 969659	0.003189	37	293	108480	1
1040	SD	Hughes County	0 925810	0 823919	0 002373	176	834	113485	0
1041	SD	Hutchinson County	0 998257	0 965653	0 093764	423	1784	110891	0
1042	SD	Hyde County	0 993553	0 989074	0.009850	111	558	127119	1
1043	SD	Jackson County	0 998868	1 137718	0 011487	51	371	70313	1

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1044	SD	Jerauld County	0 995517	0 984313	0 012952	258	712	109577	0
1045	SD	Jones County	0 998856	0 940474	0 017368	78	439	134942	1
1046	SD	Kingsbury County	0 970701	0 857380	0 029078	339	1230	115300	0
1047	SD	Lake County	0 979432	0 826147	0 023902	543	1691	108195	0
1048	SD	Lawrence County	0 999651	0 380981	0 193912	114	1093	100690	0
1049	SD	Lincoln County	0 999131	0 872414	0 036406	593	3043	100467	0
1050	SD	Lyman County	0 960696	0 806395	0 024755	93	699	106942	0
1051	SD	McCook County	0 995424	0 886536	0 104639	415	1460	108039	0
1052	SD	McPherson County	0 987057	0 909006	0 074362	179	769	110004	0
1053	SD	Marshall County	0 947181	0 904747	0 022088	442	1069	107646	0
1054	SD	Meade County	0 996605	0 934660	0 036744	54	476	102634	0
1055	SD	Mellette County	0 997509	0 838705	0 022576	65	417	89928	0
1056	SD	Miner County	0 997115	0 858626	0 054567	291	1133	108852	0
1057	SD	Minnehaha County	0 994386	0 821222	0 085211	592	2498	125480	0
1058	SD	Moody County	0 997265	0 856520	0 041193	504	2250	104111	0
1059	SD	Pennington County	0 997094	0 599962	0 034328	93	548	107152	0
1060	SD	Perkins County	0 994001	0 938692	0 014908	58	357	138580	0
1061	SD	Potter County	0 964449	0 914389	0 019439	216	966	121071	0
1062	SD	Roberts County	0 969986	0 857223	0 074028	309	1426	89264	0
1063	SD	Sanborn County	0 997897	0 886302	0 023217	382	983	100968	0
1064	SD	Shannon County	0 998681	1 057744	0 000000	20	371	49194	9
1065	SD	Spink County	0 995895	0 925365	0 010944	333	1197	143321	0
1066	SD	Stanley County	0 951427	0 978587	0 008976	49	459	114142	1
1067	SD	Sully County	0 940712	0 955040	0 001503	214	951	204862	1
1068	SD	Todd County	0 998001	1 214754	0 018073	53	460	51517	9
1069	SD	Tripp County	0 997603	0 974924	0 062380	141	754	106068	0
1070	SD	Turner County	0 998984	0 930188	0 089203	690	2319	106732	0
1071	SD	Union County	0 985654	0 880737	0 017009	741	2599	119249	0
1072	SD	Walworth County	0 951082	0 990758	0 043669	157	828	107220	0
1073	SD	Yankton County	0 979270	0 812375	0 029034	560	2052	101374	0
1074	SD	Ziebach County	0 995577	1 119733	0 012310	34	419	89982	13
1075	NE	Adams County	0 998661	0 930299	0 000703	1131	2946	121897	0
1076	NE	Antelope County	0 998411	0 892122	0 062925	614	2060	101490	0
1077	NE	Arthur County	0 995897	1 005003	0 000000	63	457	148949	9
1078	NE	Banner County	0 999794	0 853567	0 000000	219	837	134936	0
1079	NE	Blaine County	0 994969	1 012324	0 008315	97	418	121708	9
1080	NE	Boone County	0 999190	0 996158	0 015171	677	2033	110078	0
1081	NE	Box Butte County	0.997641	0 943876	0 000793	495	1316	127858	1
1082	NE	Boyd County	0 991656	0 856805	0 071082	215	847	101408	0
1083	NE	Brown County	0 996901	0 831087	0 009834	306	804	124031	0
1084	NE	Buffalo County	0 992575	0 948396	0 009289	626	2168	99905	0
1085	NE	Burt County	0 991288	0 856110	0 017300	869	2925	115205	0
1086	NE	Butler County	0 998596	0 899167	0 005994	641	2780	110369	0
1087	NE	Cass County	0 987701	0 826969	0 026867	535	3467	106719	0
1088	NE	Cedar County	0 992511	0 905003	0 065669	725	2294	96796	0
1089	NE	Chase County	0 996503	0 910722	0 000668	366	1495	131664	1
1090	NE	Cherry County	0 991827	1 019071	0 004248	64	481	113553	9
1091	NE	Cheyenne County	0 999905	1 008794	0 001570	316	889	132789	13
1092	NE	Clay County	0 999141	0 973566	0 001629	877	2916	126747	1
1093	NE	Colfax County	0 987000	0 865977	0 005876	1915	2959	103650	0
1094	NE	Cuming County	0 995580	0 944371	0 008475	3107	3195	116858	0
1095	NE	Custer County	0 999854	0 864609	0 009370	380	1083	118206	0
1096	NE	Dakota County	0 987124	0 817002	0 013076	492	2514	102831	0
1097	NE	Dawes County	0 996704	0 942357	0 005049	70	522	100101	0
1098	NE	Dawson County	0 993722	1 015873	0 000595	1211	2346	113207	9
1099	NE	Deuel County	0 997968	0 941353	0 000000	118	1066	143145	0
1100	NE	Dixon County	0 986879	0 796633	0 005835	1243	2254	105472	0
1101	NE	Dodge County	0 982540	0 873671	0 002261	962	3941	107308	0
1102	NE	Douglas County	0 974648	0 453560	0 019253	954	5023	132951	0
1103	NE	Dundy County	0 998985	0 898074	0 000635	380	1065	159834	1
1104	NE	Fillmore County	0 999725	0 922788	0 003201	800	3004	142528	0
1105	NE	Franklin County	0 999673	0 877135	0 021290	355	2231	109739	0
1106	NE	Frontier County	0 994377	0 844061	0 005827	200	1034	108933	0
1107	NE	Furnas County	0 996636	0 937729	0 004505	453	1285	123367	0
1108	NE	Gage County	0 994587	0 929372	0 100988	468	2073	106481	0
1109	NE	Garden County	0 984635	0 980627	0 000000	108	503	160295	0
1110	NE	Garfield County	0 997709	0 926819	0 001551	240	679	97019	0
1111	NE	Gosper County	0 990125	0 783306	0 002752	404	1907	136792	1
1112	NE	Grant County	0 991061	1 098600	0 000000	50	524	114054	9
1113	NE	Greeley County	0 998479	0 834010	0 024168	322	1232	112239	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1114	NE	Hall County	0 989468	0 905156	0 013092	1186	2942	109496	0
1115	NE	Hamilton County	0 994363	0 922759	0 004764	989	4081	112008	0
1116	NE	Harlan County	0 962672	0 864189	0 009679	601	1602	106177	0
1117	NE	Hayes County	0 999691	0 880729	0 002957	482	818	153887	0
1118	NE	Hitchcock County	0 988111	0 888069	0 003171	187	1083	110396	0
1119	NE	Holt County	0 997938	0 898678	0 037844	304	1086	111263	0
1120	NE	Hooker County	0 999527	0 812859	0 006343	54	314	116316	1
1121	NE	Howard County	0 989102	0 892590	0 022716	550	1763	98272	0
1122	NE	Jefferson County	0 995690	0 891148	0 078698	466	2031	118429	0
1123	NE	Johnson County	0 998237	0 775926	0 042200	325	1613	103542	0
1124	NE	Kearney County	0 999806	0 938739	0 000331	1241	3307	118964	1
1125	NE	Keith County	0 956288	0 984838	0 003000	305	864	113790	0
1126	NE	Keya Paha County	0 998857	0 901155	0 108061	114	662	142447	0
1127	NE	Kimball County	0 999388	0 829272	0 007762	100	665	135457	0
1128	NE	Knox County	0 972327	0 863857	0 059579	480	1408	94486	0
1129	NE	Lancaster County	0 990742	0 772532	0 036463	417	2837	120671	0
1130	NE	Lincoln County	0 995689	0 883864	0 010154	257	950	114548	0
1131	NE	Logan County	0 999185	0 919408	0 000000	121	590	138912	0
1132	NE	Loup County	0 997710	0 906016	0 000000	112	505	116790	0
1133	NE	McPherson County	0 998796	0 844956	0 004460	62	391	130608	1
1134	NE	Madison County	0 995708	0 878962	0 028954	784	2596	105750	0
1135	NE	Merrick County	0 980304	0 936916	0.006283	1131	2640	102628	0
1136	NE	Morrill County	0 995760	0 794996	0 000978	360	767	121301	1
1137	NE	Nance County	0 984983	0 838869	0 021511	570	1735	104016	0
1138	NE	Nemaha County	0 993718	0 862872	0 010937	488	2133	112388	0
1139	NE	Nuckolls County	0 998855	0 905674	0 035034	351	1534	106928	0
1140	NE	Otoe County	0 994731	0 826596	0 029777	444	2672	105547	0
1141	NE	Pawnee County	0 997108	0 810535	0 058544	327	1792	108073	0
1142	NE	Perkins County	0 998685	0 942805	0 008305	232	1402	175643	0
1143	NE	Phelps County	0 998836	1 087316	0 001411	1510	3428	140212	9
1144	NE	Pierce County	0 997660	0 809406	0 032340	880	2350	111446	0
1145	NE	Platte County	0 983952	0 944118	0 026502	1008	3202	110261	0
1146	NE	Polk County	0 995432	0 889869	0 004084	1171	3265	119947	0
1147	NE	Red Willow County	0 998011	0 958154	0 003638	456	1307	112252	0
1148	NE	Richardson County	0 995494	0 851191	0 064252	427	1935	100223	0
1149	NE	Rock County	0 996644	1 019301	0 011687	196	579	119256	9
1150	NE	Saline County	0 998747	0 847511	0 016969	456	2151	108649	0
1151	NE	Sarpy County	0 973225	0 682238	0 008545	1420	4441	109299	0
1152	NE	Saunders County	0 993612	0 906079	0 015564	792	3394	102180	0
1153	NE	Scotts Bluff County	0 991656	0 882740	0 002912	1132	1873	113539	0
1154	NE	Seward County	0 998331	0 856166	0 026778	784	2758	104781	0
1155	NE	Sheridan County	0 988284	0 948250	0 010570	101	544	120536	0
1156	NE	Sherman County	0 989884	0 823158	0 039266	289	1387	101632	0
1157	NE	Sioux County	0 999648	0 760480	0 000000	161	584	126164	0
1158	NE	Stanton County	0 997144	0 789617	0 021076	1105	2215	104260	0
1159	NE	Thayer County	0 998647	0 945250	0 006654	572	2215	116668	0
1160	NE	Thomas County	0 998894	0 789476	0 005735	62	429	116689	1
1161	NE	Thurston County	0 993845	0 767911	0 025391	695	2129	72270	0
1162	NE	Valley County	0 995783	0 933311	0 022706	525	1461	120292	0
1163	NE	Washington County	0 991590	0 913000	0 060558	843	3697	116276	0
1164	NE	Wayne County	0 999895	0 875562	0 034401	1062	2369	97534	0
1165	NE	Webster County	0 999776	0 835805	0 007210	744	1481	121599	0
1166	NE	Wheeler County	0 999289	0 717078	0 018053	1150	811	243408	0
1167	NE	York County	0 999377	0 937765	0 000682	1083	4231	118958	1
1168	KS	Allen County	0 995695	0 878562	0 147362	254	1227	92366	0
1169	KS	Anderson County	0 997647	1 014506	0 042863	299	1419	97554	9
1170	KS	Atchison County	0 993784	0 885764	0 054754	360	1854	101070	0
1171	KS	Barber County	0 998153	0 880774	0 022753	193	929	125331	0
1172	KS	Barton County	0 992829	1 014000	0 008687	588	1450	118131	9
1173	KS	Bourbon County	0 997283	0 827182	0 053333	198	1115	100033	0
1174	KS	Brown County	0 997342	0 928506	0 039091	558	2151	107660	0
1175	KS	Butler County	0 987325	0 837716	0 008335	403	1413	117522	0
1176	KS	Chase County	0 997272	0 708699	0 009891	324	1128	120346	0
1177	KS	Chautauqua County	0 995128	0 941986	0 014408	190	815	92298	0
1178	KS	Cherokee County	0 993487	0 721189	0 013596	327	1605	86372	0
1179	KS	Cheyenne County	0 999069	0 907232	0 000000	215	1044	129246	0
1180	KS	Clark County	0 997421	0 906120	0 000812	362	714	164155	1
1181	KS	Clay County	0 982323	0 924499	0 031218	308	1414	105635	0
1182	KS	Cloud County	0 996050	0 889604	0 015657	204	1432	108168	0
1183	KS	Coffey County	0 962781	0 876050	0 008080	292	1420	108378	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1184	KS	Comanche County	0 998242	0 965208	0 023633	157	724	121586	0
1185	KS	Cowley County	0 994411	0 870642	0 018980	263	1332	104402	0
1186	KS	Crawford County	0 996423	0 798007	0 044861	274	1332	105946	0
1187	KS	Decatur County	0 999291	0 919852	0 003829	391	1056	174857	0
1188	KS	Dickinson County	0 995054	0 947391	0 040485	381	1474	110633	0
1189	KS	Doniphan County	0 987687	0 803938	0 022556	455	2069	95570	0
1190	KS	Douglas County	0 963040	0 759179	0 086169	386	2628	95746	0
1191	KS	Edwards County	0 999870	1 013217	0 002058	479	1382	137767	9
1192	KS	Elk County	0 996078	0 781575	0 007335	153	848	99390	0
1193	KS	Ellis County	0 999443	0 950472	0 048434	235	1092	113708	0
1194	KS	Ellsworth County	0 989582	0 965438	0 003359	112	1101	103027	1
1195	KS	Finney County	0 998058	0 895771	0 000200	1111	1489	103901	1
1196	KS	Ford County	0 999283	0 954688	0 001321	1195	1302	117380	0
1197	KS	Franklin County	0 995153	0 861201	0 105419	316	2025	97785	0
1198	KS	Geary County	0 951286	0 667171	0 065328	278	1322	82369	0
1199	KS	Gove County	0 999916	0 979246	0 006238	514	1066	143362	0
1200	KS	Graham County	0 999573	0 891805	0 020769	187	878	110999	0
1201	KS	Grant County	0 999682	0 928421	0 00465	2089	1479	144459	1
1202	KS	Gray County	0 999494	0 930763	0 000509	1260	1500	123462	1
1203	KS	Greeley County	1 000000	0 851692	0 000708	552	1254	185518	1
1204	KS	Greenwood County	0 988858	0 827646	0 026787	205	950	105770	0
1205	KS	Hamilton County	0 998868	0 835535	0 001202	518	853	158669	1
1206	KS	Harper County	0 998116	0 973021	0 005646	264	1376	127573	0
1207	KS	Harvey County	0 997944	0 925997	0 049903	446	2140	103996	0
1208	KS	Haskell County	0 999376	0 992513	0 000371	2440	1891	129869	1
1209	KS	Hodgeman County	0 999631	0 871939	0 004273	558	887	120759	0
1210	KS	Jackson County	0 998413	0 808863	0 069866	235	1494	101408	0
1211	KS	Jefferson County	0 962603	0 791844	0 081989	326	1945	105242	0
1212	KS	Jewell County	0 994210	0 833176	0 015720	256	1335	112198	0
1213	KS	Johnson County	0 992924	0 463350	0 044899	494	4621	177160	0
1214	KS	Kearny County	0 998251	0 929145	0 000331	968	1183	151462	1
1215	KS	Kingman County	0 996505	0 984242	0 037592	220	1422	108432	0
1216	KS	Kiowa County	0 999682	0 864768	0 001930	215	1120	132457	1
1217	KS	Labette County	0 993187	0 834381	0 030158	461	1428	103942	0
1218	KS	Lane County	0 999703	0 913679	0 000000	738	1034	125182	0
1219	KS	Leavenworth County	0 989203	0 696520	0 192995	382	2777	97304	0
1220	KS	Lincoln County	0 998513	1 048550	0 023314	165	1132	111047	9
1221	KS	Linn County	0 987493	0 714596	0 074521	224	1433	93877	0
1222	KS	Logan County	0 999901	0 878290	0 007355	112	799	110112	1
1223	KS	Lyon County	0 995070	0 891709	0 019775	320	1284	101598	0
1224	KS	McPherson County	0 998363	0 934068	0 047768	481	2012	118145	0
1225	KS	Marion County	0 989061	0 974219	0 104808	320	1506	100467	0
1226	KS	Marshall County	0 998027	0 991878	0 043473	307	1635	112489	0
1227	KS	Meade County	0 998733	0 951897	0 002091	266	1092	121782	1
1228	KS	Miami County	0 977228	0 777485	0 071888	280	2839	103779	0
1229	KS	Mitchell County	0 973931	1 070019	0 009850	362	1450	118795	9
1230	KS	Montgomery County	0 990495	0 784007	0 045824	234	1480	101774	0
1231	KS	Morris County	0 992233	0 918181	0 041320	293	1285	98604	0
1232	KS	Morton County	0 999971	0 914855	0 000486	143	1003	118592	0
1233	KS	Nemaha County	0 999504	0 959170	0 106661	437	1852	113471	0
1234	KS	Neosho County	0 989319	0 892608	0 074773	259	1409	106001	0
1235	KS	Ness County	0 999760	0 971711	0 008523	124	847	148645	0
1236	KS	Norton County	0 995977	0 828576	0 004305	203	1117	113952	0
1237	KS	Osage County	0 978181	0 775686	0 015555	247	1480	99078	0
1238	KS	Osborne County	0 998101	0 958160	0 018102	175	941	119608	0
1239	KS	Ottawa County	0 998945	0 824182	0 012216	281	1410	99248	0
1240	KS	Pawnee County	0 999491	0 930505	0 001390	797	1264	124071	1
1241	KS	Phillips County	0 990408	1 026177	0 013607	195	985	116303	9
1242	KS	Pottawatomie County	0 979319	0 835320	0 025230	281	1428	98049	0
1243	KS	Pratt County	0 998964	0 919029	0 000968	821	1548	125243	0
1244	KS	Rawlins County	0 999899	0 936474	0 020164	129	903	132789	0
1245	KS	Reno County	0 986766	0 872940	0 058641	388	1875	116479	0
1246	KS	Republic County	0 994711	0 966638	0 007129	610	1669	109611	0
1247	KS	Rice County	0 997642	0 930475	0 005436	470	1494	111907	0
1248	KS	Riley County	0 979890	0 584820	0 045757	291	1503	88783	0
1249	KS	Rooks County	0 992164	1 017086	0 035100	152	929	103549	9
1250	KS	Rush County	0 999708	0 929913	0 015828	135	1024	125866	0
1251	KS	Russell County	0 984101	0 818906	0 035997	115	950	121240	0
1252	KS	Saline County	0 997690	0 875591	0 056146	213	1617	128704	0
1253	KS	Scott County	0 999836	1 054787	0 000000	2067	1252	156231	9

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1254	KS	Sedgwick County	0 990858	0 797176	0 179726	337	2557	130560	0
1255	KS	Seward County	0 998457	0 801553	0 000000	1641	1430	116228	0
1256	KS	Shawnee County	0 988315	0 646052	0 036272	322	2485	132470	0
1257	KS	Sheridan County	0 999724	0 933159	0 015997	330	1282	116391	0
1258	KS	Sherman County	0 999799	0 917700	0 001883	284	1352	126571	1
1259	KS	Smith County	0 998769	0 937817	0 012438	248	1130	109800	0
1260	KS	Stafford County	0 996639	0 860536	0 000000	523	1417	134495	0
1261	KS	Stanton County	0 999917	0 946110	0 000503	801	1391	176076	1
1262	KS	Stevens County	0 999754	0 968145	0 000670	1098	1317	167420	1
1263	KS	Sumner County	0 997492	0 909020	0 037173	241	1559	112171	0
1264	KS	Thomas County	0 999897	1 021278	0 000000	351	1461	123787	9
1265	KS	Trego County	0 988066	0 851451	0 016401	316	1083	112354	0
1266	KS	Wabaunsee County	0 997000	0 828930	0 035033	232	989	103677	0
1267	KS	Wallace County	0 999942	0 806252	0 002205	160	1080	115104	1
1268	KS	Washington County	0 999578	0 906211	0 045168	329	1620	99519	0
1269	KS	Wichita County	0 999967	0 965030	0 000000	1849	1381	159678	0
1270	KS	Wilson County	0 998031	0 851395	0 007721	247	1307	97040	0
1271	KS	Woodson County	0 990414	0 830090	0 000729	229	1009	93538	0
1272	KS	Wyandotte County	0 972370	0 232758	0 013749	535	5243	95732	1
1273	DE	Kent County	0 738196	0 522125	0 083485	1399	5687	101232	0
1274	DE	New Castle County	0 863773	0 319369	0 073593	1143	8849	150934	0
1275	DE	Sussex County	0 784189	0 507715	0 013127	3307	6214	112428	0
1276	MD	Allegany County	0 989772	0 138865	0 353833	224	3026	100149	0
1277	MD	Anne Arundel County	0 707515	0 162729	0 015845	641	9964	154409	1
1278	MD	Baltimore County	0 877594	0 217262	0 152840	1206	13905	158967	0
1279	MD	Calvert County	0 623547	0 270976	0 000000	450	10646	148286	0
1280	MD	Caroline County	0 981852	0 619727	0 045242	1655	5490	95644	0
1281	MD	Carroll County	0 992770	0 547917	0 317609	1051	9686	149024	0
1282	MD	Cecil County	0 833137	0 360109	0 223721	1093	8412	126957	0
1283	MD	Charles County	0 716787	0 201254	0 010464	414	8410	130838	0
1284	MD	Dorchester County	0 567244	0 346813	0 002777	1280	5373	115361	1
1285	MD	Frederick County	0 993326	0 525042	0 620383	1211	9403	135152	0
1286	MD	Garrett County	0 987908	0 266865	0 547585	456	3239	89217	0
1287	MD	Harford County	0 836028	0 345275	0 364608	730	9986	141838	0
1288	MD	Howard County	0 994540	0 276485	0 156948	1049	11199	186575	0
1289	MD	Kent County	0 674459	0 734067	0 174232	1025	7058	130459	0
1290	MD	Montgomery County	0 976988	0 260546	0 163402	830	14078	203745	0
1291	MD	Prince George's County	0 973938	0 174940	0 024308	997	10383	132538	1
1292	MD	Queen Anne's County	0 730119	0 694076	0 092130	825	6861	144567	0
1293	MD	St. Mary's County	0 472558	0 335132	0 033152	521	6819	113126	0
1294	MD	Somerset County	0 535720	0 265762	0 006921	4568	6099	80527	1
1295	MD	Talbot County	0 564547	0 633328	0 051407	804	8968	170800	0
1296	MD	Washington County	0 979875	0 422654	0 590580	1163	7277	109983	0
1297	MD	Wicomico County	0 943472	0 377996	0 001081	4459	7287	112306	1
1298	MD	Worcester County	0 681152	0 354990	0 002711	3018	5266	128386	1
1299	MD	Baltimore city	0 877610	0 000000	0 000000	0	0	116926	0
1300	DC	District of Columbia	0 898382	0 000000	0 000000	0	0	159868	0
1301	VA	Accomack County	0 346984	0 314718	0 000000	1938	4386	100386	0
1302	VA	Albemarle County	0 995200	0 407658	0 123163	286	7559	136798	23
1303	VA	Alleghany County	0 997671	0 090445	0 070519	196	3420	98354	24
1304	VA	Amelia County	0 995144	0 308317	0 132636	1362	3533	103149	0
1305	VA	Amherst County	0 992576	0 296866	0 161980	146	2652	85024	1
1306	VA	Appomattox County	0 996869	0 368453	0 365510	216	4061	98990	0
1307	VA	Arlington County	0 996373	0 010060	0 000000	155	3077	204835	2
1308	VA	Augusta County	0 999413	0 462204	0 227345	955	5152	108480	23
1309	VA	Bath County	0 994921	0 138101	0 081127	93	2522	124681	1
1310	VA	Bedford County	0 981134	0 415042	0 406847	256	3355	111169	23
1311	VA	Bland County	0 999823	0 356209	0 306144	200	2054	85844	0
1312	VA	Botetourt County	0 993995	0 278795	0 349988	320	3746	118274	0
1313	VA	Brunswick County	0 994324	0 233691	0 120647	462	2490	86887	0
1314	VA	Buchanan County	1 000000	0 026751	0 460800	179	3031	94445	14
1315	VA	Buckingham County	0 995368	0 177620	0 050029	581	2566	88431	0
1316	VA	Campbell County	0 994521	0 416482	0 233164	274	2592	113526	23
1317	VA	Caroline County	0 988283	0 151386	0 034749	397	3868	108459	1
1318	VA	Carroll County	0 997271	0 371055	0 184521	431	3149	85776	23
1319	VA	Charles City County	0 894919	0 246843	0 000000	462	4148	123448	0
1320	VA	Charlotte County	0 994915	0 371503	0 196346	319	2479	89582	0
1321	VA	Chesterfield County	0 970979	0 063835	0 000000	997	7316	134922	0
1322	VA	Clarke County	0 991197	0 604415	0 238838	642	8896	153460	0
1323	VA	Craig County	0 999986	0 215158	0 123596	140	2403	90063	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1324	VA	Culpeper County	0 997001	0 472621	0 296438	516	5896	117847	0
1325	VA	Cumberland County	0 995871	0 322784	0 067594	1091	3078	90178	0
1326	VA	Dickenson County	0 994288	0 040697	0 022077	155	3773	91641	15
1327	VA	Dinwiddie County	0 993528	0 266602	0 110966	484	3012	109211	23
1328	VA	Essex County	0 901571	0 341132	0 000000	424	3321	113803	0
1329	VA	Fairfax County	0 972373	0 062069	0 043204	524	13797	196416	24
1330	VA	Fauquier County	0 997666	0 565916	0 298232	396	8374	165076	0
1331	VA	Floyd County	0 999965	0 477212	0 228122	453	3135	99011	0
1332	VA	Fluvanna County	0 990349	0 317066	0 031648	193	3147	100826	1
1333	VA	Franklin County	0 972627	0 375832	0 658034	596	3899	98157	0
1334	VA	Frederick County	0 997403	0 369883	0 059776	596	7722	110999	23
1335	VA	Giles County	0 991492	0 319123	0 071058	137	2022	98232	1
1336	VA	Gloucester County	0 751995	0 176588	0 030625	475	7322	107084	1
1337	VA	Goochland County	0 980843	0 282569	0 185014	228	5299	160532	0
1338	VA	Grayson County	0 992679	0 481178	0 265199	306	3257	84889	0
1339	VA	Greene County	0 997654	0 369579	0 310774	295	5155	94520	0
1340	VA	Greensville County	0 995378	0 271684	0 018378	754	2194	105526	24
1341	VA	Halifax County	0 987557	0 447078	0 027101	354	2307	97527	23
1342	VA	Hanover County	0 997155	0 318215	0 152681	537	6632	144913	0
1343	VA	Henrico County	0 979079	0 158796	0 000000	700	7328	161771	0
1344	VA	Henry County	0 994701	0 200097	0 072162	201	2418	112157	24
1345	VA	Highland County	1 000000	0 364097	0 029361	250	2353	87259	1
1346	VA	Isle of Wight County	0 870727	0 426609	0 000000	1102	4725	121450	0
1347	VA	James City County	0 795163	0 104511	0 118616	628	8074	116546	24
1348	VA	King and Queen County	0 969157	0 259396	0 047793	425	2955	98882	1
1349	VA	King George County	0 958495	0 327898	0 000000	343	4653	118707	0
1350	VA	King William County	0 964223	0 336529	0 167369	494	3808	119046	0
1351	VA	Lancaster County	0 575503	0 230672	0 005376	327	3357	153440	0
1352	VA	Lee County	0 999573	0 462141	0 037032	294	2772	77601	0
1353	VA	Loudoun County	0 997651	0 587461	0 122411	338	10973	180722	0
1354	VA	Louisa County	0 973883	0 255742	0 126833	300	4073	101523	0
1355	VA	Lunenburg County	0 998564	0 309762	0 002253	384	2559	84834	0
1356	VA	Madison County	0 998884	0 488999	0 307339	460	4964	97900	0
1357	VA	Mathews County	0 340081	0 112172	0 000000	971	7242	126950	0
1358	VA	Mecklenburg County	0 918510	0 420329	0 115123	469	2490	96078	0
1359	VA	Middlesex County	0 618245	0 257898	0 075969	436	5320	122737	1
1360	VA	Montgomery County	0 996912	0 398088	0 412118	379	4260	89697	23
1361	VA	Nelson County	0 995881	0 240407	0 024832	197	3915	102512	1
1362	VA	New Kent County	0 937724	0 136767	0 000000	458	5039	141750	0
1363	VA	Northampton County	0 260726	0 395324	0 000000	1260	5093	96430	0
1364	VA	Northumberland County	0 673182	0 335474	0 000000	476	4085	120048	0
1365	VA	Nottoway County	0 995571	0 317687	0 137639	952	3433	102133	0
1366	VA	Orange County	0 995040	0 492434	0 186157	465	5023	118605	0
1367	VA	Page County	0 990554	0 325692	0 016466	2682	6161	99857	0
1368	VA	Patrick County	0 994397	0 256930	0 248414	348	3129	102892	0
1369	VA	Pittsylvania County	0 992475	0 478098	0 067601	466	2942	100894	23
1370	VA	Powhatan County	0 995721	0 258207	0 451047	454	5764	120956	0
1371	VA	Prince Edward County	0 996994	0 303741	0 240860	426	3437	82274	0
1372	VA	Prince George County	0 943389	0 287599	0 000000	388	3780	90822	23
1373	VA	Prince William County	0 969556	0 152234	0 444188	422	11973	139142	23
1374	VA	Pulaski County	0 972611	0.349973	0 282857	401	3837	90666	0
1375	VA	Rappahannock County	0 999026	0 462873	0 014582	193	7694	131563	0
1376	VA	Richmond County	0 884835	0 317763	0 017161	532	3541	103779	1
1377	VA	Roanoke County	0 999290	0 155361	0 027111	1276	6968	132971	23
1378	VA	Rockbridge County	0 997752	0 369385	0 246921	251	3979	99464	23
1379	VA	Rockingham County	0 997486	0 433344	0 143494	3883	7827	106847	23
1380	VA	Russell County	0 995565	0 529888	0 075828	300	2405	87882	0
1381	VA	Scott County	0 996205	0 390735	0 076005	299	3373	81441	0
1382	VA	Shenandoah County	0 999370	0 382494	0 093539	1103	5928	115822	0
1383	VA	Smyth County	0 999517	0 414631	0 278144	406	2938	96965	0
1384	VA	Southampton County	0 995360	0 464556	0 000000	777	3297	90984	23
1385	VA	Spotsylvania County	0 972344	0 206192	0 389865	277	8258	112706	23
1386	VA	Stafford County	0 965849	0 116357	0 077088	230	10475	147060	1
1387	VA	Surry County	0 899402	0 295418	0 015266	883	4339	121606	1
1388	VA	Sussex County	0 995734	0 263421	0 000000	807	2762	97297	0
1389	VA	Tazewell County	0 999716	0 416446	0 158660	242	2631	98929	0
1390	VA	Warren County	0 987877	0 284883	0 039066	258	5942	103596	0
1391	VA	Washington County	0 994041	0 526354	0 245090	618	4280	103393	23
1392	VA	Westmoreland County	0 907156	0 383738	0 028475	666	4440	114914	1
1393	VA	Wise County	0 997048	0 051314	0 334047	174	4028	105018	23

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1394	VA	Wythe County	0 997045	0 443075	0 400944	482	3172	93267	0
1395	VA	York County	0 490095	0 034885	0 050157	3009	19563	140117	24
1396	VA	Alexandria city	0 993195	0 000000	0 000000	0	0	214704	0
1397	VA	Bedford city	0 997917	0 000000	0 000000	0	0	111169	23
1398	VA	Bristol city	1 000000	0 000000	0 000000	0	0	103393	23
1399	VA	Buena Vista city	1 000000	0 000000	0 000000	0	0	99464	23
1400	VA	Charlottesville city	1 000000	0 000000	0 000000	0	0	136798	23
1401	VA	Chesapeake city	0 970831	0 248996	0 058490	1075	6459	114630	0
1402	VA	Clifton Forge city	0 998628	0 000000	0 000000	0	0	98354	23
1403	VA	Colonial Heights city	0 958395	0 000000	0 000000	0	0	109211	23
1404	VA	Covington city	1 000000	0 000000	0 000000	0	0	98354	23
1405	VA	Danville city	0 980046	0 000000	0 000000	0	0	100894	23
1406	VA	Emporia city	0 990230	0 000000	0 000000	0	0	105526	23
1407	VA	Fairfax city	1 000000	0 000000	0 000000	0	0	196416	23
1408	VA	Falls Church city	1 000000	0 000000	0 000000	0	0	196416	23
1409	VA	Franklin city	0 993951	0 000000	0 000000	0	0	90984	23
1410	VA	Fredericksburg city	0 999120	0 000000	0 000000	0	0	112706	23
1411	VA	Galax city	1 000000	0 000000	0 000000	0	0	85776	23
1412	VA	Hampton city	0 403078	0 000000	0 000000	0	0	99857	0
1413	VA	Harrisonburg city	0 998332	0 000000	0 000000	0	0	106847	23
1414	VA	Hopewell city	0 955812	0 000000	0 000000	0	0	90822	23
1415	VA	Lexington city	1 000000	0 000000	0 000000	0	0	99464	23
1416	VA	Lynchburg city	0 992653	0 000000	0 000000	0	0	113526	23
1417	VA	Manassas city	0 997768	0 000000	0 000000	0	0	139142	23
1418	VA	Manassas Park city	1 000000	0 000000	0 000000	0	0	139142	23
1419	VA	Martinsville city	0 995265	0 000000	0 000000	0	0	112157	23
1420	VA	Newport News city	0 574113	0 000000	0 000000	0	0	103813	0
1421	VA	Norfolk city	0 558239	0 000000	0 000000	0	0	111162	0
1422	VA	Norton city	1 000000	0 000000	0 000000	0	0	105018	23
1423	VA	Petersburg city	0 985284	0 000000	0 000000	0	0	109211	23
1424	VA	Poquoson city	0 197859	0 000000	0 000000	0	0	140117	23
1425	VA	Portsmouth city	0 710980	0 000000	0 000000	0	0	100264	0
1426	VA	Radford city	0 963547	0 000000	0 000000	0	0	89697	23
1427	VA	Richmond city	0 960401	0 000000	0 000000	0	0	157253	0
1428	VA	Roanoke city	0 999290	0 000000	0 000000	0	0	128474	0
1429	VA	Salem city	1 000000	0 000000	0 000000	0	0	132971	23
1430	VA	South Boston city	0 997566	0 000000	0 000000	0	0	97527	23
1431	VA	Staunton city	1 000000	0 000000	0 000000	0	0	108480	23
1432	VA	Suffolk city	0 932382	0 324338	0 003886	1103	4833	113837	1
1433	VA	Virginia Beach city	0 499294	0 272653	0 008470	970	6611	119865	1
1434	VA	Waynesboro city	0 998682	0 000000	0 000000	0	0	108480	23
1435	VA	Williamsburg city	0 985241	0 000000	0 000000	0	0	116546	23
1436	VA	Winchester city	1 000000	0 000000	0 000000	0	0	110999	23
1437	WV	Barbour County	0 994269	0 348802	0 133532	120	1723	65937	0
1438	WV	Berkeley County	0 998659	0 357233	0 176248	722	5063	90388	0
1439	WV	Boone County	0 999609	0 007862	0 000000	77	3589	97670	0
1440	WV	Braxton County	0 994794	0 223360	0 026685	65	1057	84340	1
1441	WV	Brooke County	0 963024	0 214101	0 621622	165	1713	90923	0
1442	WV	Cabell County	0 977686	0 197451	0 069942	157	2202	106888	0
1443	WV	Calhoun County	0 999944	0 194419	0 088148	42	1197	60634	1
1444	WV	Clay County	0 995867	0 067843	0 093496	82	2429	59293	0
1445	WV	Doddridge County	0 999848	0 288572	0 000000	69	1513	67380	0
1446	WV	Fayette County	0 993365	0 048136	0 041361	151	2339	76767	1
1447	WV	Gilmer County	0 999932	0 242357	0 000000	87	1330	68687	0
1448	WV	Grant County	0 993783	0 348112	0 011592	529	2218	90768	0
1449	WV	Greenbrier County	0 996903	0 274975	0 078917	468	2246	88648	0
1450	WV	Hampshire County	0 995508	0 330081	0 010084	187	2534	83615	1
1451	WV	Hancock County	0 939819	0 145065	0 000000	282	3820	113234	0
1452	WV	Hardy County	0 998144	0 379599	0 015646	1178	3428	91323	0
1453	WV	Harrison County	0 998704	0 332605	0 184268	116	2333	97961	0
1454	WV	Jackson County	0 987671	0 339513	0 076737	101	2134	78583	0
1455	WV	Jefferson County	0 990416	0 553742	0 415489	690	7269	96809	0
1456	WV	Kanawha County	0 991333	0 034524	0 055734	231	2700	116445	0
1457	WV	Lewis County	0 997566	0 325875	0 055301	113	1598	78867	0
1458	WV	Lincoln County	0 997366	0 107207	0 031980	133	2624	64406	1
1459	WV	Logan County	0 996936	0 010708	0 631010	130	2035	89711	1
1460	WV	McDowell County	0 999661	0 003179	0.000000	343	2777	68152	0
1461	WV	Marion County	0 994130	0 206040	0 070017	89	2572	96464	1
1462	WV	Marshall County	0 983429	0 327394	0 393981	139	1844	89298	0
1463	WV	Mason County	0 970889	0 423895	0 363624	301	2210	82403	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1464	WV	Mercer County	0 999510	0 210148	0 018434	123	2605	102052	1
1465	WV	Mineral County	0 995757	0 356391	0 209848	131	2110	86961	0
1466	WV	Mingo County	0 997562	0 000954	0 000000	699	4784	87198	0
1467	WV	Monongalia County	0 987102	0 236290	0 143192	116	2939	102804	0
1468	WV	Monroe County	0 999376	0 491281	0 218020	267	2190	80344	0
1469	WV	Morgan County	0 997073	0 149217	0 097079	181	3644	89271	1
1470	WV	Nicholas County	0 991181	0 079694	0 022468	172	3095	80303	1
1471	WV	Ohio County	0 975393	0 311426	0 569444	235	2230	123144	0
1472	WV	Pendleton County	0 999702	0 398818	0 004616	703	1980	72893	0
1473	WV	Pleasants County	0 971309	0 187043	0 061379	133	2244	88600	1
1474	WV	Pocahontas County	0 998347	0 191895	0 054925	117	2066	87029	0
1475	WV	Preston County	0 995295	0 334215	0 326210	179	1955	69317	0
1476	WV	Putnam County	0 988102	0 251924	0 000000	137	2477	93680	0
1477	WV	Raleigh County	0 996049	0 084005	0 060731	129	2896	89860	1
1478	WV	Randolph County	0 999808	0 156572	0 164878	133	2183	83324	0
1479	WV	Ritchie County	0 999853	0 244464	0 013667	132	1433	75982	1
1480	WV	Roane County	0 999634	0 265423	0 000000	73	1551	67353	0
1481	WV	Summers County	0 982202	0 249687	0 095761	139	2420	61575	1
1482	WV	Taylor County	0 983455	0 378258	0 059971	255	2114	68965	1
1483	WV	Tucker County	0 994688	0 119709	0 080408	99	2627	81522	1
1484	WV	Tyler County	0 987964	0.287333	0 292419	87	1359	86352	0
1485	WV	Upshur County	0 999714	0 258423	0 049030	119	3213	72764	0
1486	WV	Wayne County	0 987455	0 088418	0 202966	93	1852	80195	0
1487	WV	Webster County	0 999919	0 026227	0 212061	65	2474	65077	1
1488	WV	Wetzel County	0 994065	0 161494	0 048251	55	1435	91398	0
1489	WV	Wirt County	0 992168	0 240304	0 155836	229	1815	57037	0
1490	WV	Wood County	0 974479	0 254548	0 128701	105	2975	105784	0
1491	WV	Wyoming County	0 998120	0 017759	0 389044	115	2919	68809	1
1492	NC	Alamance County	0 990543	0 366683	0 267643	794	5175	117217	0
1493	NC	Alexander County	0 988735	0 317955	0 193832	1643	4522	101564	0
1494	NC	Alleghany County	0 996420	0 483510	0 594266	566	4160	87713	0
1495	NC	Anson County	0 989653	0 207790	0 012200	2172	3981	93152	0
1496	NC	Ashe County	0 998482	0 382872	0 086198	264	3822	84448	0
1497	NC	Avery County	0 999177	0 124686	0 059798	782	6938	84780	1
1498	NC	Beaufort County	0 863896	0 272865	0 003465	921	3293	93646	1
1499	NC	Bertie County	0 943365	0 379892	0 000000	1102	3382	91255	0
1500	NC	Bladen County	0 986319	0 228139	0 006521	1107	2948	82220	1
1501	NC	Brunswick County	0 814141	0 072500	0 000000	724	4085	84848	0
1502	NC	Buncombe County	0 994291	0 222808	0 366309	587	9317	114697	0
1503	NC	Burke County	0 984139	0 097654	0 033928	1566	6168	102248	0
1504	NC	Cabarrus County	0 998376	0 270402	0 097152	488	6107	112902	0
1505	NC	Caldwell County	0 994418	0 103302	0 157119	1000	5576	101225	0
1506	NC	Camden County	0 787308	0 279503	0 000000	954	3190	100399	0
1507	NC	Carteret County	0 393150	0 188283	0 000000	489	5057	97663	0
1508	NC	Caswell County	0 993608	0 460340	0 066234	413	2481	77133	0
1509	NC	Catawba County	0 967257	0 245533	0 261952	573	6842	117827	0
1510	NC	Chatham County	0 963198	0 247873	0 055529	2112	4772	117468	0
1511	NC	Cherokee County	0.975385	0 082137	0 063161	1133	5548	83114	0
1512	NC	Chowan County	0 739922	0 487847	0 000000	1115	3567	95881	0
1513	NC	Clay County	0 973276	0 119386	0 197198	699	5895	81224	0
1514	NC	Cleveland County	0 991527	0 316206	0 084838	650	4162	106712	0
1515	NC	Columbus County	0 982388	0 271247	0 013655	831	3537	83818	1
1516	NC	Craven County	0 913020	0 198552	0 004113	1268	3594	100196	1
1517	NC	Cumberland County	0 991842	0 235719	0 000000	869	4219	87530	0
1518	NC	Currituck County	0 497844	0 249266	0 000000	855	4289	95793	0
1519	NC	Dare County	0 244412	0 028845	0 000000	194	1734	103264	0
1520	NC	Davidson County	0 974279	0 260851	0 242615	548	5867	102133	0
1521	NC	Davie County	0 993697	0 404857	0 347081	495	4057	124214	0
1522	NC	Duplin County	0 998394	0 474803	0 000522	4307	3916	82247	0
1523	NC	Durham County	0.974891	0 105783	0 056230	834	11251	130303	1
1524	NC	Edgecombe County	0 996952	0 558097	0 002551	1002	2910	91039	1
1525	NC	Forsyth County	0 992138	0 182454	0 050672	692	10621	139833	0
1526	NC	Franklin County	0 994835	0 375982	0 007971	978	3962	86223	1
1527	NC	Gaston County	0 979889	0 152147	0 595739	535	4321	110532	0
1528	NC	Gates County	0 985546	0 296013	0 000000	1132	3471	99072	0
1529	NC	Graham County	0 968397	0 047515	0 301211	345	5806	62469	1
1530	NC	Granville County	0 989880	0 458973	0 076712	490	3334	91621	0
1531	NC	Greene County	0 998269	0 660976	0 000000	2599	4980	86297	0
1532	NC	Guilford County	0 988449	0 273152	0 129361	850	8300	137828	0
1533	NC	Halifax County	0 992440	0 440340	0 009988	1038	2587	86941	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1534	NC	Harnett County	0 989586	0 335226	0 027950	1198	4927	81326	0
1535	NC	Haywood County	0 998269	0 197357	0 275243	508	6402	96416	0
1536	NC	Henderson County	0 997317	0 218512	0 197487	2005	8485	123103	0
1537	NC	Hertford County	0 980146	0 333538	0 000000	1264	3142	86528	0
1538	NC	Hoke County	0 997155	0 226410	0 000000	1392	2709	68795	0
1539	NC	Hyde County	0 430354	0 238975	0 000000	657	2618	84536	0
1540	NC	Iredell County	0 967951	0 402959	0 362296	1061	5954	115910	0
1541	NC	Jackson County	0 992346	0 042392	0 063093	665	6417	87171	0
1542	NC	Johnston County	0 995073	0 454560	0 006567	1219	4817	100609	1
1543	NC	Jones County	0 996962	0 226391	0 000000	1045	3505	81333	0
1544	NC	Lee County	0 992004	0 227338	0 011945	1031	4318	112930	1
1545	NC	Lenoir County	0 994439	0 553931	0 003768	1727	4519	101131	1
1546	NC	Lincoln County	0 973171	0 305287	0 349539	597	4138	103610	0
1547	NC	McDowell County	0 989397	0 075058	0 127522	1010	5549	84435	0
1548	NC	Macon County	0 994248	0 066829	0 346570	434	6527	95455	0
1549	NC	Madison County	0 995199	0 324427	0 054390	273	5144	82565	0
1550	NC	Martin County	0 999331	0 445076	0 000000	1071	3473	96633	0
1551	NC	Mecklenburg County	0 959416	0 082657	0 069948	3110	15308	141844	0
1552	NC	Mitchell County	0 996948	0 162314	0 118213	339	4583	82552	1
1553	NC	Montgomery County	0 979971	0 117650	0 029543	2527	3642	87984	0
1554	NC	Moore County	0 989330	0 194503	0 000468	2248	4463	129050	0
1555	NC	Nash County	0 995580	0 517777	0 001499	1718	4162	111805	1
1556	NC	New Hanover County	0 606659	0 027379	0 000000	1797	16076	115950	0
1557	NC	Northampton County	0 974341	0 452343	0 005220	1138	2452	93612	1
1558	NC	Onslow County	0 844141	0 130384	0 015897	2266	4214	69019	1
1559	NC	Orange County	0 996610	0 263776	0 220202	1234	7760	134495	0
1560	NC	Pamlico County	0 595059	0 204037	0 000000	943	3142	91878	0
1561	NC	Pasquotank County	0 783872	0 573079	0 000000	837	3706	89711	0
1562	NC	Pender County	0 933557	0 116886	0 005017	1411	3729	90544	1
1563	NC	Perquimans County	0 751440	0 434491	0 000000	1063	4265	85410	0
1564	NC	Person County	0 970901	0 461386	0 035384	549	3110	94994	0
1565	NC	Pitt County	0 995125	0 465221	0 000000	1708	4653	107369	0
1566	NC	Polk County	0 996872	0 152021	0 310368	316	4695	136182	0
1567	NC	Randolph County	0 996829	0 287428	0 148553	1756	5162	99708	0
1568	NC	Richmond County	0 988325	0 171122	0 000000	1825	2791	87875	0
1569	NC	Robeson County	0 997714	0 481069	0 001728	913	2859	73482	1
1570	NC	Rockingham County	0 989727	0 361003	0 055793	588	3957	101855	0
1571	NC	Rowan County	0 976157	0 320424	0 339602	616	4251	106278	0
1572	NC	Rutherford County	0 996889	0 153168	0 126165	216	3423	96166	0
1573	NC	Sampson County	0 997894	0 439685	0 007039	3298	3891	94852	0
1574	NC	Scotland County	0 995347	0 262846	0 000000	1182	2532	89264	0
1575	NC	Stanly County	0 977249	0 352218	0 073925	1245	3892	99762	0
1576	NC	Stokes County	0 991119	0 362166	0 031211	651	4245	96565	0
1577	NC	Surry County	0 997670	0 353537	0 098157	1196	4885	104714	0
1578	NC	Swain County	0 976839	0 017447	0 092836	842	6647	67258	1
1579	NC	Transylvania County	0 994193	0 051159	0 000000	1243	8714	104544	0
1580	NC	Tyrrell County	0 649549	0 271271	0 000000	828	2626	76185	0
1581	NC	Union County	0 996527	0 410294	0 005721	3556	5969	122520	0
1582	NC	Vance County	0 939586	0 417306	0 030721	665	3565	88634	1
1583	NC	Wake County	0 972494	0 224571	0 033989	1362	8276	139921	0
1584	NC	Warren County	0 966043	0 318794	0 037489	843	2632	80181	0
1585	NC	Washington County	0 819891	0 462421	0 000000	1285	3185	85336	0
1586	NC	Watauga County	0 999346	0 233586	0 039698	400	5849	86250	0
1587	NC	Wayne County	0 992597	0 507695	0 004486	2669	4564	90422	0
1588	NC	Wilkes County	0 996399	0 237517	0 020867	3367	5486	99559	0
1589	NC	Wilson County	0 991467	0 599208	0 000000	1557	4565	106062	0
1590	NC	Yadkin County	0 994338	0 483130	0 186029	1027	5208	103366	0
1591	NC	Yancey County	0 997840	0 192001	0 109561	351	5305	81739	0
1592	SC	Abbeville County	0 994028	0 276594	0 218125	184	2752	79064	0
1593	SC	Aiken County	0 993100	0 198851	0 023476	590	3194	121227	1
1594	SC	Allendale County	0 989440	0 311009	0 000000	397	1799	73442	0
1595	SC	Anderson County	0 947944	0 349602	0 159713	550	4794	96213	0
1596	SC	Bamberg County	0 994391	0 347065	0 281193	372	2079	71213	0
1597	SC	Barnwell County	0 984227	0 212891	0 021953	329	2671	95319	1
1598	SC	Beaufort County	0 635997	0 119245	0 000000	346	2474	123320	0
1599	SC	Berkeley County	0 894502	0 072501	0 008678	610	4217	87489	1
1600	SC	Calhoun County	0 969306	0 373841	0 013023	456	2233	83182	1
1601	SC	Charleston County	0 675997	0 055168	0 000000	1274	8681	108473	0
1602	SC	Cherokee County	0 988441	0 263257	0 022415	364	2961	99451	1
1603	SC	Chester County	0 990397	0 253506	0 205646	196	2347	80208	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1604	SC	Chesterfield County	0 991237	0 214492	0 003803	1363	2653	88309	0
1605	SC	Clarendon County	0 872851	0 349336	0 000000	670	2301	70543	0
1606	SC	Colleton County	0 933243	0 186898	0 000000	241	2805	80215	0
1607	SC	Darlington County	0 990326	0 436041	0 025012	600	2556	90984	0
1608	SC	Dillon County	0 995892	0 420057	0 000000	790	2858	78258	0
1609	SC	Dorchester County	0 996607	0 168834	0 051003	541	3580	95096	0
1610	SC	Edgefield County	0 990836	0 214361	0 091487	610	3227	80628	0
1611	SC	Fairfield County	0 967011	0 126792	0 107373	90	2233	89406	1
1612	SC	Florence County	0 995219	0 380887	0 004282	648	2683	101334	1
1613	SC	Georgetown County	0 787232	0 071498	0 000000	631	3021	99288	0
1614	SC	Greenville County	0 993833	0 131789	0 219012	489	7555	121064	0
1615	SC	Greenwood County	0 983976	0 241054	0 011585	332	2771	102641	1
1616	SC	Hampton County	0 995052	0 271352	0 000000	317	2726	86345	0
1617	SC	Horry County	0 903357	0 269712	0 001853	804	3853	99519	0
1618	SC	Jasper County	0 954051	0 173126	0 000000	131	2310	77634	0
1619	SC	Kershaw County	0 981081	0 113972	0 000000	928	2687	101368	0
1620	SC	Lancaster County	0 988652	0 165986	0 020663	448	3652	92644	1
1621	SC	Laurens County	0 987723	0 280715	0 255296	317	3612	94039	0
1622	SC	Lee County	0 997565	0 518450	0 008093	490	2011	67454	1
1623	SC	Lexington County	0 922749	0 184236	0 008119	1937	5363	122771	0
1624	SC	McCormick County	0 912939	0 084670	0 000000	356	2698	74349	0
1625	SC	Marion County	0 989873	0 250401	0 004473	770	3675	79687	1
1626	SC	Marlboro County	0 988510	0 341546	0 000000	493	1968	76090	0
1627	SC	Newberry County	0 974489	0 232762	0 291120	926	3227	93247	0
1628	SC	Oconee County	0 927959	0 174713	0 037480	1057	4603	106217	0
1629	SC	Orangeburg County	0 980639	0 370274	0 180212	598	2380	88241	0
1630	SC	Pickens County	0 970788	0 138982	0 048397	253	5425	100941	1
1631	SC	Richland County	0 980298	0 137045	0 000000	298	4312	116072	0
1632	SC	Saluda County	0 979884	0 383131	0 109200	880	3005	86785	0
1633	SC	Spartanburg County	0 989965	0 206263	0 154725	538	5427	106854	0
1634	SC	Sumter County	0 975633	0 325368	0 010574	922	2457	79944	1
1635	SC	Union County	0 996616	0 170138	0 164369	88	3206	80608	1
1636	SC	Williamsburg County	0 996832	0 289728	0 005379	580	2725	69459	1
1637	SC	York County	0 980999	0 276433	0 110317	449	4200	113200	0
1638	GA	Appling County	0 993412	0 324118	0 059700	728	2138	87828	1
1639	GA	Atkinson County	0 982508	0 358891	0 000000	1217	2378	91486	0
1640	GA	Bacon County	0 996611	0 431721	0 019993	583	2076	85525	1
1641	GA	Baker County	0 982964	0 495454	0 007779	542	2264	87313	1
1642	GA	Baldwin County	0 966265	0 199329	0 147374	189	2319	99112	1
1643	GA	Banks County	0 999145	0 330287	0 003171	3045	7477	94188	0
1644	GA	Barrow County	0 996135	0 345316	0 001540	2775	9277	97006	0
1645	GA	Bartow County	0 977287	0 289057	0 045767	769	4623	94872	0
1646	GA	Ben Hill County	0 991317	0 292049	0 013037	748	3173	91282	1
1647	GA	Berrien County	0 988373	0 446204	0 010791	658	2378	86670	1
1648	GA	Bibb County	0 979738	0 106906	0 180791	511	4183	118856	0
1649	GA	Bleckley County	0 992105	0 452643	0 028522	511	2144	96735	1
1650	GA	Brantley County	0 993262	0 096898	0 014354	1160	3517	71688	1
1651	GA	Brooks County	0 991798	0 534398	0 068744	637	2582	71545	0
1652	GA	Bryan County	0 971873	0 056409	0 000000	303	2509	87740	0
1653	GA	Bulloch County	0 990825	0 489728	0 000187	618	2652	82538	0
1654	GA	Burke County	0 994619	0 313249	0 172070	438	1717	76049	0
1655	GA	Butts County	0 982093	0 244594	0 049743	316	3902	97473	1
1656	GA	Calhoun County	0 987993	0 634955	0 000000	686	2307	97595	0
1657	GA	Camden County	0 805027	0 044508	0 520145	49	1723	75765	1
1658	GA	Candler County	0 992546	0 361049	0 011795	682	2952	86670	1
1659	GA	Carroll County	0 990887	0 258342	0 020731	1377	4554	97961	0
1660	GA	Catoosa County	0 997297	0 283648	0 149633	1497	5081	87103	0
1661	GA	Charlton County	0 997192	0 043418	0.000000	427	2838	80831	0
1662	GA	Chatham County	0 681377	0.030222	0.000000	619	3913	120400	0
1663	GA	Chattahoochee County	0 990467	0 037060	0.000000	139	1423	78061	0
1664	GA	Chattooga County	0 999303	0.262204	0 152036	207	2097	80378	0
1665	GA	Cherokee County	0 976318	0 124045	0 012188	3453	11850	122155	0
1666	GA	Clarke County	0 996202	0 149473	0 010438	3803	6212	102506	1
1667	GA	Clay County	0 899600	0 341583	0 034419	625	2144	68125	1
1668	GA	Clayton County	0 988609	0 049499	0 000000	344	8454	109529	0
1669	GA	Clinch County	0 981941	0 026184	0 000000	555	3050	73855	0
1670	GA	Cobb County	0 987472	0 046808	0 000000	3818	13542	148557	10
1671	GA	Coffee County	0 994116	0 466464	0 013657	1291	2775	90571	0
1672	GA	Colquitt County	0 992312	0 560665	0 030325	1031	2812	94547	0
1673	GA	Columbia County	0 942287	0 145370	0 085849	198	6052	113756	1

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1674	GA	Cook County	0 982302	0 495370	0 000000	735	2596	84455	0
1675	GA	Coweta County	0 993384	0 148019	0 183670	309	5186	108473	0
1676	GA	Crawford County	0 995886	0 182485	0 032166	751	2042	84543	1
1677	GA	Crisp County	0 973761	0 627218	0 000000	677	2405	84787	0
1678	GA	Dade County	0 998958	0 231802	0 032193	552	4465	72663	1
1679	GA	Dawson County	0 986444	0 141111	0 006690	3598	8654	90172	1
1680	GA	Decatur County	0 957694	0 441372	0 016125	675	3119	91743	1
1681	GA	DeKalb County	0 990213	0 017740	0 000000	522	11524	142271	0
1682	GA	Dodge County	0 994930	0 303404	0 019448	485	1789	85098	1
1683	GA	Dooly County	0 989461	0 623482	0 007813	749	2223	90192	1
1684	GA	Dougherty County	0 985130	0 337126	0 000000	454	3351	101774	0
1685	GA	Douglas County	0 995219	0 063899	0 167290	337	6598	116614	1
1686	GA	Early County	0 990249	0 562742	0 000000	542	2399	93206	0
1687	GA	Echols County	0 960382	0 063257	0 000000	845	2673	75508	0
1688	GA	Effingham County	0 992960	0 142649	0 071201	294	2946	84584	1
1689	GA	Elbert County	0 984563	0 229775	0 327900	475	2804	96992	0
1690	GA	Emanuel County	0 993577	0 281771	0 062421	357	1623	82653	1
1691	GA	Evans County	0 989966	0 342930	0 029294	771	2538	101422	1
1692	GA	Fannin County	0 985466	0 063092	0 042462	1181	5866	83398	0
1693	GA	Fayette County	0 990603	0 175822	0 194137	213	7790	157890	1
1694	GA	Floyd County	0 989943	0 224228	0 042031	593	3436	108791	1
1695	GA	Forsyth County	0 912786	0 250904	0 025004	3543	12573	138546	1
1696	GA	Franklin County	0 988510	0 442912	0 007463	2719	5408	99972	0
1697	GA	Fulton County	0 988966	0 064945	0 183309	456	9559	154558	1
1698	GA	Gilmer County	0 988109	0 092919	0 026245	4211	6714	103196	0
1699	GA	Glascock County	0 998212	0 309188	0 000000	137	2011	98726	0
1700	GA	Glynn County	0 724453	0 035811	0 000000	66	3542	118572	0
1701	GA	Gordon County	0 993071	0 324989	0 017226	2208	4580	100819	0
1702	GA	Grady County	0 995189	0 469399	0 043241	989	3779	79849	0
1703	GA	Greene County	0 955919	0 188079	0 351799	855	3000	92190	0
1704	GA	Gwinnett County	0 991121	0 087495	0 014730	1285	12853	134523	1
1705	GA	Habersham County	0 996395	0 202605	0 006068	6289	11250	102445	1
1706	GA	Hall County	0 917191	0 214110	0 031729	5684	8886	114758	0
1707	GA	Hancock County	0 988569	0 116816	0 281564	92	1534	79362	1
1708	GA	Haralson County	0 996383	0 174592	0 052402	1080	3938	90327	0
1709	GA	Harris County	0 980508	0 104572	0 000000	166	3507	104416	0
1710	GA	Hart County	0 905601	0 393803	0 128360	1192	3902	103718	0
1711	GA	Heard County	0 983219	0 127942	0 000000	958	3742	75399	0
1712	GA	Henry County	0 994437	0 220916	0 134494	411	7380	121965	0
1713	GA	Houston County	0 991903	0 304450	0 108419	629	3926	106454	0
1714	GA	Irwin County	0 983684	0 592216	0 005230	649	2653	79660	1
1715	GA	Jackson County	0 998175	0 379106	0 008359	2630	6424	103820	0
1716	GA	Jasper County	0 991602	0 256480	0 103772	1104	2421	98645	0
1717	GA	Jeff Davis County	0 993921	0 340350	0 013883	455	1830	95617	1
1718	GA	Jefferson County	0 996388	0 402941	0 165544	388	1903	86297	0
1719	GA	Jenkins County	0 992517	0 346279	0 264663	573	1851	78610	0
1720	GA	Johnson County	0 993094	0 366359	0 029201	220	1332	80533	1
1721	GA	Jones County	0 995866	0 124572	0 512936	681	3162	91513	0
1722	GA	Lamar County	0 994786	0 335697	0 224577	669	3821	95624	0
1723	GA	Lanier County	0 934847	0 342535	0 000000	981	2366	87909	0
1724	GA	Laurens County	0 992756	0 323123	0 066943	359	2006	95638	0
1725	GA	Lee County	0 982781	0 460089	0 022614	581	2360	86392	1
1726	GA	Liberty County	0 861492	0 046904	0 000000	45	1486	69737	0
1727	GA	Lincoln County	0 820457	0 241701	0 150034	281	2478	84441	1
1728	GA	Long County	0 993826	0 046633	0 000000	745	2662	54727	0
1729	GA	Lowndes County	0 987470	0 226267	0 037969	512	3456	95976	0
1730	GA	Lumpkin County	0 998603	0 127883	0 020684	3811	9016	97202	1
1731	GA	McDuffie County	0 975499	0 203194	0 169521	854	3792	92407	0
1732	GA	McIntosh County	0 754165	0 028848	0 000000	40	1515	84441	0
1733	GA	Macon County	0 993433	0 468138	0 388938	1124	2287	84787	0
1734	GA	Madison County	0 995946	0 339240	0 010538	2286	4885	88438	0
1735	GA	Marion County	0 998709	0 193466	0 047435	639	1872	76937	1
1736	GA	Meriwether County	0 995974	0 213347	0 237847	166	3206	79334	0
1737	GA	Miller County	0 997621	0 671164	0 011243	671	2429	87591	1
1738	GA	Mitchell County	0 996465	0 627325	0 118425	1010	2708	78962	0
1739	GA	Monroe County	0 994553	0 176124	0 225886	790	3311	97629	0
1740	GA	Montgomery County	0 991956	0 413322	0 003597	339	1643	86805	0
1741	GA	Morgan County	0 986017	0 415854	0 313216	1296	2979	109997	0
1742	GA	Murray County	0 992909	0 149471	0 091927	757	3957	83981	1
1743	GA	Muscogee County	0 978690	0 035178	0 000000	263	7323	105018	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1744	GA	Newton County	0 990124	0 259119	0 196769	577	4944	107050	0
1745	GA	Oconee County	0 997932	0 435993	0 020467	2373	5498	110322	0
1746	GA	Oglethorpe County	0 997605	0 195903	0 087739	1669	3407	93159	0
1747	GA	Paulding County	0.995155	0 092905	0 060237	1226	7438	92204	1
1748	GA	Peach County	0 997448	0 459895	0 146019	1160	3771	112435	0
1749	GA	Pickens County	0 997210	0 122861	0 005909	4254	7963	106813	1
1750	GA	Pierce County	0 998126	0 368530	0 169642	611	2925	77059	0
1751	GA	Pike County	0 995366	0 325109	0 151025	289	3574	88079	0
1752	GA	Polk County	0 996815	0.231055	0 065730	699	4550	86473	0
1753	GA	Pulaski County	0 990131	0 507756	0 008606	663	2433	99979	1
1754	GA	Putnam County	0 955200	0 157592	0 852644	1373	2521	81475	0
1755	GA	Quitman County	0 941725	0 119171	0 000000	370	1771	86697	0
1756	GA	Rabun County	0 984244	0 053616	0 006267	1889	7276	85728	0
1757	GA	Randolph County	0 996187	0 348970	0 025312	567	2284	80391	1
1758	GA	Richmond County	0 988508	0 077019	0 120267	478	3775	107972	1
1759	GA	Rockdale County	0 989147	0 153458	0 144395	248	6231	130682	1
1760	GA	Schley County	0 998739	0 353508	0 070833	414	1928	82470	0
1761	GA	Sc生生 County	0 989149	0 334410	0 034270	386	2247	81278	1
1762	GA	Seminole County	0 927800	0 715203	0 012272	686	2683	87124	1
1763	GA	Spalding County	0 991780	0 190098	0 200261	381	8770	99790	1
1764	GA	Stephens County	0 973029	0 135282	0 027250	2170	4252	91350	1
1765	GA	Stewart County	0 990240	0 167043	0 019540	479	1952	73029	1
1766	GA	Sumter County	0 985254	0 547287	0 044997	652	2298	90991	0
1767	GA	Talbot County	0 996134	0 152231	0 101360	118	1872	82098	1
1768	GA	Taliaferro County	0 999615	0 154440	0 787627	416	2111	107213	0
1769	GA	Tattnall County	0 990736	0 387188	0 005077	1508	2951	91973	1
1770	GA	Taylor County	0 994291	0 224997	0 000000	616	2025	88356	0
1771	GA	Telfair County	0 993355	0 251805	0 000000	431	1948	81014	0
1772	GA	Terrell County	0 993563	0 665213	0 012717	505	2002	73902	1
1773	GA	Thomas County	0 993320	0 495783	0 046404	416	2872	100149	0
1774	GA	Tift County	0 986074	0 674691	0 007116	1126	3673	102919	1
1775	GA	Toombs County	0 994733	0 378388	0 005789	485	2512	99627	0
1776	GA	Towns County	0 968888	0 092988	0 003373	444	7844	83771	0
1777	GA	Treutlen County	0 992626	0 255323	0 000000	338	2333	71498	0
1778	GA	Troup County	0 928154	0 153940	0 326729	180	2336	102025	0
1779	GA	Turner County	0 986932	0 539747	0 009727	859	3082	78894	0
1780	GA	Twiggs County	0 992974	0 135108	0 058726	251	1776	76727	1
1781	GA	Union County	0 980330	0 106386	0 016411	3399	8380	78535	10
1782	GA	Upson County	0 993464	0 157748	0 000000	626	3449	89501	0
1783	GA	Walker County	0 998892	0 310999	0 118569	444	3908	89061	0
1784	GA	Walton County	0 997534	0 264704	0 017179	771	5943	92515	0
1785	GA	Ware County	0 995568	0 093298	0 039019	655	2447	93362	1
1786	GA	Warren County	0 995756	0 257186	0 510281	266	1863	76334	0
1787	GA	Washington County	0 994218	0 256719	0 142085	282	1709	99607	0
1788	GA	Wayne County	0 993666	0 131944	0 076115	441	2473	92535	0
1789	GA	Webster County	0 996570	0.397361	0 000000	441	1856	94391	0
1790	GA	Wheeler County	0 991926	0 255867	0 026313	358	1679	74431	1
1791	GA	White County	0 997699	0 156025	0 018265	4682	9490	102126	0
1792	GA	Whitfield County	0 997719	0 208453	0 024143	2243	5549	120969	0
1793	GA	Wilcox County	0 992332	0 474505	0 022775	548	2030	90056	0
1794	GA	Wilkes County	0 994470	0 308511	0 219326	406	2526	101544	0
1795	GA	Wilkinson County	0 987968	0 111392	0 000000	95	1235	97399	0
1796	GA	Worth County	0 991663	0 548586	0 029790	769	2468	76883	0
1797	FL	Alachua County	0 902099	0 341595	0 155847	513	4182	108764	0
1798	FL	Baker County	0 993762	0 065381	0 261540	2807	4877	88431	0
1799	FL	Bay County	0 739094	0 018689	0 865022	225	4042	100338	1
1800	FL	Bradford County	0 976974	0 193106	0 127604	1153	4809	80168	1
1801	FL	Brevard County	0 654033	0 306402	0 040917	435	3669	119669	1
1802	FL	Broward County	0 916005	0 030678	0 157936	3617	14064	151415	0
1803	FL	Calhoun County	0 987824	0 119280	0 049212	865	3077	64176	0
1804	FL	Charlotte County	0 807267	0 511774	0 018965	412	3146	110823	1
1805	FL	Citrus County	0.754754	0 189213	0 000000	194	4688	93064	0
1806	FL	Clay County	0 933845	0 223600	0 479524	976	5547	108568	0
1807	FL	Collier County	0 878688	0 232954	0 000000	2134	5128	173319	0
1808	FL	Columbia County	0 995060	0 190065	0 174170	526	3537	84922	1
1809	FL	Dade County	0 800324	0 067243	0 002014	10541	23985	120719	1
1810	FL	DeSoto County	0 996524	0 820376	0 005587	950	5289	82531	1
1811	FL	Dixie County	0 815152	0 070335	0 536843	209	2007	69622	1
1812	FL	Duval County	0 842713	0 080844	0 374772	1385	7611	119628	0
1813	FL	Escambia County	0 742382	0 134624	0 250303	676	3807	104253	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1814	FL	Flagler County	0 849741	0 168353	0 000000	479	3759	90097	0
1815	FL	Franklin County	0 520224	0 010386	0 000000	55	355	79097	12
1816	FL	Gadsden County	0 976623	0 175131	0 000000	3035	5363	87347	0
1817	FL	Gilchrist County	0 981353	0 317940	0 623361	982	3982	78217	0
1818	FL	Glades County	0 784338	0 747340	0 025353	379	1787	62110	1
1819	FL	Gulf County	0 747652	0 039274	0 000000	643	3591	84184	10
1820	FL	Hamilton County	0 991390	0 210622	0 099527	514	2244	70861	1
1821	FL	Hardee County	0 998395	0 803148	0 141004	963	5335	96593	0
1822	FL	Hendry County	0 968744	0 718198	0 002986	1275	4848	97900	0
1823	FL	Hernando County	0 811877	0 199337	0 319143	756	5974	94276	0
1824	FL	Highlands County	0 929609	0 735057	0 085350	950	5355	100609	0
1825	FL	Hillsborough County	0 829956	0 394620	0 067556	2413	10108	114305	0
1826	FL	Holmes County	0 987300	0 280757	0 106968	893	2407	76700	0
1827	FL	Indian River County	0 815675	0 542329	0 014866	2052	9349	164535	1
1828	FL	Jackson County	0 959259	0 416629	0 131038	533	2307	85451	0
1829	FL	Jefferson County	0 938917	0 309349	0 136621	439	3736	90835	1
1830	FL	Lafayette County	0 990638	0 275849	0 506761	1127	2678	83331	0
1831	FL	Lake County	0 824122	0 326397	0 068936	1120	8285	118504	0
1832	FL	Lee County	0 663037	0 207505	0 008814	1888	7616	127858	1
1833	FL	Leon County	0 950081	0 236120	0 374100	94	4463	111229	1
1834	FL	Levy County	0 791870	0 266209	0 489003	552	2477	78874	0
1835	FL	Liberty County	0 991350	0 021940	0 000000	136	3049	77120	0
1836	FL	Madison County	0 966565	0 298537	0 072288	372	2082	74200	1
1837	FL	Manatee County	0 830162	0 631784	0 053492	1730	5217	121897	0
1838	FL	Marion County	0 949425	0 293148	0 128227	534	6459	93484	0
1839	FL	Martin County	0 738088	0 536477	0 026961	2008	7869	183723	0
1840	FL	Monroe County	0 266832	0 000050	0 000000	158632	256628	133784	10
1841	FL	Nassau County	0 897667	0 107814	0 343399	1529	4152	122527	0
1842	FL	Okaloosa County	0 864795	0 094680	0 115698	271	3445	107037	1
1843	FL	Okeechobee County	0 868102	0 710056	0 606845	936	3926	86731	0
1844	FL	Orange County	0 903732	0 238290	0 013838	3709	10615	121098	1
1845	FL	Osceola County	0 877552	0 846882	0 004042	183	2635	111426	0
1846	FL	Palm Beach County	0 852424	0 489974	0 003226	3452	9200	181508	1
1847	FL	Pasco County	0 858267	0 463997	0 182480	695	5521	94445	0
1848	FL	Pinellas County	0 460971	0 022995	0 000000	6868	18616	146369	0
1849	FL	Polk County	0 932712	0 509466	0 043005	822	6591	100480	0
1850	FL	Putnam County	0 873010	0 228522	0 086300	779	3602	77587	1
1851	FL	St Johns County	0 741407	0 125296	0 000000	2319	5885	133053	0
1852	FL	St Lucie County	0 831950	0 820487	0 003471	1703	7798	95753	1
1853	FL	Santa Rosa County	0 879261	0 121930	0 033496	669	3208	91878	1
1854	FL	Sarasota County	0 788331	0 413317	0 076055	309	4858	168978	1
1855	FL	Seminole County	0 893667	0 302349	0 105716	845	7223	121139	1
1856	FL	Sumter County	0 940249	0 725389	0 072292	360	3640	82823	0
1857	FL	Suwannee County	0 993846	0 367939	0 133985	1420	3666	82057	0
1858	FL	Taylor County	0 845704	0 110887	0 546641	88	1555	95062	6
1859	FL	Union County	0 962262	0 313932	0 088265	417	3396	66581	1
1860	FL	Volusia County	0 771986	0 195275	0 009113	1410	7221	105987	1
1861	FL	Wakulla County	0 824557	0 022352	0 474164	432	4052	88661	1
1862	FL	Walton County	0 854250	0 142899	0 096895	569	2459	78488	1
1863	FL	Washington County	0 941615	0 121830	0 278970	601	2462	76713	0
1864	KY	Adair County	0 986916	0 683048	0 349087	397	2459	79944	0
1865	KY	Allen County	0 983186	0 706858	0 066713	387	2127	78793	0
1866	KY	Anderson County	0 992106	0 694062	0 153306	375	3682	98211	0
1867	KY	Ballard County	0 917981	0 696188	0 053777	650	2709	105425	0
1868	KY	Barren County	0 982012	0 791284	0 260856	570	3113	88614	0
1869	KY	Bath County	0 984100	0 743533	0 096800	456	2482	76016	0
1870	KY	Bell County	0 998387	0 023469	0 000000	88	3006	72663	0
1871	KY	Boone County	0 958328	0 513058	0 065937	427	7038	123780	0
1872	KY	Bourbon County	0 999210	1 109124	0 011170	948	4961	100223	9
1873	KY	Boyd County	0 989844	0 271518	0 000000	400	3026	113952	0
1874	KY	Boyle County	0 994028	0 931622	0 085739	598	4186	98394	0
1875	KY	Bracken County	0 972839	0 761193	0 162328	438	2448	76036	0
1876	KY	Breathitt County	0 999809	0 134414	0 016608	118	1516	71329	1
1877	KY	Breckinridge County	0 977595	0 728069	0 012070	319	2194	83879	0
1878	KY	Bullitt County	0 996399	0 318168	0 146152	339	4396	95028	0
1879	KY	Butler County	0 992021	0 513942	0 038657	246	1958	77011	0
1880	KY	Caldwell County	0 996563	0 579995	0 071410	369	2010	82823	0
1881	KY	Calloway County	0 940284	0 555530	0 072431	601	2856	95136	0
1882	KY	Campbell County	0 950607	0 447899	0 053167	269	7875	108486	0
1883	KY	Carlisle County	0 967169	0 640965	0 090423	562	2636	85498	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1884	KY	Carroll County	0 947697	0 730338	0 028705	355	2613	97073	0
1885	KY	Carter County	0 996247	0 429341	0 047581	220	2204	70238	0
1886	KY	Casey County	0 999739	0 673861	0 219884	327	2390	69364	0
1887	KY	Christian County	0 996282	0 648342	0 010129	491	2584	77208	0
1888	KY	Clark County	0 996685	0 890228	0 009462	553	4255	99830	0
1889	KY	Clay County	0 999852	0 226811	0 025617	196	1943	58879	0
1890	KY	Clinton County	0 960655	0 596710	0 155835	318	2641	60112	0
1891	KY	Crittenden County	0 976252	0 539867	0 052575	183	1562	75569	0
1892	KY	Cumberland County	0 983834	0 555725	0 097621	177	1393	67055	0
1893	KY	Daviess County	0 970883	0 845154	0 028735	661	4279	107166	0
1894	KY	Edmonson County	0 982495	0 477509	0 185515	345	2185	53664	0
1895	KY	Elliott County	0 994775	0 402623	0 000000	180	2022	53373	0
1896	KY	Estill County	0 993353	0 426339	0 019494	185	2300	68091	0
1897	KY	Fayette County	0 996455	0 808124	0 000537	2117	8495	130858	1
1898	KY	Fleming County	0 998800	0 862797	0 283024	525	2458	77966	0
1899	KY	Floyd County	0 997056	0 043266	0 000000	138	3968	85146	0
1900	KY	Franklin County	0 992232	0 638957	0 012317	475	3947	114989	0
1901	KY	Fulton County	0.906313	0 724030	0 000000	526	2882	95502	0
1902	KY	Gallatin County	0 944217	0 653701	0 073508	367	3577	94026	0
1903	KY	Garrard County	0 988593	0 932963	0 065211	550	2853	90273	0
1904	KY	Grant County	0 996698	0 764523	0 037247	312	3052	87476	0
1905	KY	Graves County	0 998482	0 591297	0 036247	801	2843	97094	0
1906	KY	Grayson County	0 985967	0 639310	0 201512	362	2348	73313	0
1907	KY	Green County	0 999829	0 729505	0 238861	454	2309	82206	0
1908	KY	Greenup County	0 976568	0 453414	0 051009	215	2339	94838	0
1909	KY	Hancock County	0 949113	0 576895	0 001904	391	2430	98679	0
1910	KY	Hardin County	0 997033	0 562797	0 165118	430	3512	91160	0
1911	KY	Harlan County	0 998334	0 016234	0 162806	106	2125	78637	1
1912	KY	Harrison County	0 999433	0 899085	0 027510	461	2938	89101	0
1913	KY	Hart County	0 995257	0 752997	0 234189	440	2445	74404	0
1914	KY	Henderson County	0 941974	0 702267	0 000845	501	3144	108879	1
1915	KY	Henry County	0 993846	0 863843	0 162599	537	3024	93091	0
1916	KY	Hickman County	0 966458	0 633173	0 027640	541	2398	88539	0
1917	KY	Hopkins County	0 993366	0 411012	0 001681	344	2667	114697	1
1918	KY	Jackson County	0 999304	0 364030	0 155884	321	2407	56834	0
1919	KY	Jefferson County	0 966149	0 181396	0 052751	731	9258	129639	0
1920	KY	Jessamine County	0 992549	0 889188	0 011364	647	5663	104179	0
1921	KY	Johnson County	0 990911	0 137766	0 018930	164	3200	83127	0
1922	KY	Kenton County	0 985457	0 424694	0 109589	310	5107	114704	0
1923	KY	Knott County	0 997678	0 014302	0 236809	110	1978	62212	1
1924	KY	Knox County	0 999898	0 186693	0 069869	159	2933	67332	0
1925	KY	Larue County	0 998883	0 717412	0 276716	487	3046	92976	0
1926	KY	Laurel County	0 981808	0 356927	0 089133	375	3880	81746	0
1927	KY	Lawrence County	0 996903	0 180958	0 031760	168	1763	66073	0
1928	KY	Lee County	0 993575	0 154881	0 048830	165	1603	59523	1
1929	KY	Leslie County	0 999182	0 005805	0 000000	110	2042	67251	0
1930	KY	Letcher County	0 999783	0 015590	0 000000	45	2471	73774	0
1931	KY	Lewis County	0 977374	0 515047	0 211947	234	1839	71207	0
1932	KY	Lincoln County	0 999369	0 807312	0.193152	553	3344	69872	0
1933	KY	Livingston County	0 923707	0 589286	0 064868	222	1557	87354	0
1934	KY	Logan County	0 997464	0 783540	0 104401	543	2780	88959	0
1935	KY	Lyon County	0 841240	0 358550	0 022985	304	2258	79389	0
1936	KY	McCracken County	0 936758	0 390502	0 060829	508	3120	118192	0
1937	KY	McCreary County	0 993179	0 050729	0 068829	88	2223	51903	1
1938	KY	McLean County	0 992667	0 830544	0 004917	587	3177	90090	0
1939	KY	Madison County	0 994529	0 876651	0 021881	461	3395	85830	0
1940	KY	Magoffin County	0 999980	0 227886	0 012798	145	1873	59895	1
1941	KY	Marion County	0 999393	0 791173	0 259091	504	2970	84035	0
1942	KY	Marshall County	0 895939	0 390179	0 009399	468	2925	98868	1
1943	KY	Martin County	0 999523	0 035596	0 000000	287	3239	81698	0
1944	KY	Mason County	0 977728	0 934760	0 232200	543	3217	95658	0
1945	KY	Meade County	0 951562	0 605355	0 041207	369	2936	77397	0
1946	KY	Menifee County	0 989658	0 326749	0 099379	271	2846	55581	0
1947	KY	Mercer County	0 991321	0 829228	0 161601	662	3704	107640	0
1948	KY	Metcalfe County	0 999762	0 735128	0 282599	436	2161	71478	0
1949	KY	Monroe County	0 996095	0 783625	0 341494	369	2244	82687	0
1950	KY	Montgomery County	0 998936	0 892031	0 065209	476	2850	89508	0
1951	KY	Morgan County	0 993633	0 430533	0 020673	220	1880	64325	0
1952	KY	Muhlenberg County	0 990131	0 387938	0 012408	520	2154	82443	0
1953	KY	Nelson County	0 996652	0 706059	0 284644	527	3258	98753	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
1954	KY	Nicholas County	0 998786	0 893288	0 020060	398	1934	90510	0
1955	KY	Ohio County	0 995069	0 420466	0 005997	325	2227	77214	0
1956	KY	Oldham County	0 962576	0 697280	0 151166	501	6771	138620	0
1957	KY	Owen County	0 994326	0 784540	0 103900	302	2371	78901	0
1958	KY	Owsley County	1 000000	0 281672	0 018204	228	2234	57918	0
1959	KY	Pendleton County	0 995286	0 710842	0 072023	296	3312	81881	0
1960	KY	Perry County	0 998578	0 020408	0 000000	187	2531	91865	0
1961	KY	Pike County	0 998529	0 012215	0 000000	102	2569	88790	0
1962	KY	Powell County	0 999617	0 287571	0 000000	210	2831	66940	0
1963	KY	Pulaski County	0 977179	0 515165	0 197751	421	3602	87659	0
1964	KY	Robertson County	0 999630	0 827861	0 071722	318	2257	67468	0
1965	KY	Rockcastle County	0 998310	0 456536	0 141908	289	2767	71796	0
1966	KY	Rowan County	0 981034	0 276581	0 036883	239	3302	70421	0
1967	KY	Russell County	0 896430	0 563051	0 220093	601	3487	86196	0
1968	KY	Scott County	0 999568	0 844191	0 006318	739	5000	121897	0
1969	KY	Shelby County	0 996182	0 934732	0 222315	643	4624	115849	0
1970	KY	Simpson County	0 999982	0 779039	0 059353	597	2871	89677	0
1971	KY	Spencer County	0 969272	0 789071	0 279336	579	3476	86995	0
1972	KY	Taylor County	0 973947	0 745341	0 181535	524	2915	89650	0
1973	KY	Todd County	0 998356	0 685037	0 103411	712	2622	78136	0
1974	KY	Trigg County	0 921522	0 394241	0 005451	412	1973	89332	1
1975	KY	Trimble County	0 952737	0 748646	0 071915	387	3131	95414	0
1976	KY	Union County	0 949562	0 890564	0 000677	628	3501	100406	1
1977	KY	Warren County	0 995473	0 724505	0 131149	533	3739	99884	0
1978	KY	Washington County	0 996926	0 859642	0 198754	482	2606	88411	0
1979	KY	Wayne County	0 948778	0 462030	0 076723	419	2019	64528	0
1980	KY	Webster County	0 997084	0 655553	0 001385	430	2452	114352	1
1981	KY	Whitley County	0 988786	0 158134	0 033808	156	2633	83920	0
1982	KY	Wolfe County	0 999652	0 428805	0 004400	175	1969	60898	0
1983	KY	Woodford County	0 993209	1 013209	0 000895	1512	6792	159123	13
1984	TN	Anderson County	0 978792	0 193965	0 089798	333	5603	111094	0
1985	TN	Bedford County	0 997541	0 704543	0 158948	684	2957	97615	0
1986	TN	Benton County	0 905054	0 249284	0 045419	170	2357	89982	1
1987	TN	Bledsoe County	0 999026	0 357995	0 203942	307	2570	70218	0
1988	TN	Blount County	0 985732	0 269048	0 238596	447	6500	103657	0
1989	TN	Bradley County	0 991657	0 436580	0 199393	1144	5004	99329	0
1990	TN	Campbell County	0 963500	0 098613	0 203886	315	4555	73909	0
1991	TN	Cannon County	0 999760	0 567851	0 222589	349	2489	92027	0
1992	TN	Carroll County	0 998608	0 431792	0 049544	323	2085	86880	0
1993	TN	Carter County	0 981098	0 167825	0 191027	472	5555	77147	0
1994	TN	Cheatham County	0 985433	0 300892	0 015075	287	3786	94405	0
1995	TN	Chester County	0 999233	0 387259	0 068754	243	1889	78095	0
1996	TN	Claiborne County	0 983537	0 513508	0 083469	414	4047	86467	0
1997	TN	Clay County	0 910727	0 466420	0 115330	222	2218	81488	0
1998	TN	Cocke County	0 980261	0 302230	0 178788	436	3934	73110	0
1999	TN	Coffee County	0 987146	0 482285	0 229275	547	3355	105107	0
2000	TN	Crockett County	0 999221	0 853541	0 005331	629	3103	89948	1
2001	TN	Cumberland County	0.995125	0 222064	0 202056	412	3653	84123	0
2002	TN	Davidson County	0 954574	0 147207	0 049406	312	8367	132111	1
2003	TN	Decatur County	0 968063	0 406435	0 000000	164	1725	77492	0
2004	TN	DeKalb County	0 925796	0 491535	0 079836	477	2967	88174	0
2005	TN	Dickson County	0 997113	0 460112	0 089401	196	3118	101334	0
2006	TN	Dyer County	0 969737	0 706621	0 017758	523	2871	100609	0
2007	TN	Fayette County	0 997553	0 572765	0 045014	479	2619	90382	0
2008	TN	Fentress County	0 999327	0 220770	0 058702	660	2669	65138	0
2009	TN	Franklin County	0 963024	0 382675	0 180160	759	4123	79903	0
2010	TN	Gibson County	0 998483	0 705448	0 017215	567	2661	95333	0
2011	TN	Giles County	0 999593	0 655393	0 195019	290	2622	91994	0
2012	TN	Grainger County	0 926918	0 582168	0 110868	368	3435	71146	0
2013	TN	Greene County	0 996118	0 595286	0 312200	549	4633	88167	0
2014	TN	Grundy County	0 998398	0 185180	0 042587	1346	3625	74641	0
2015	TN	Hamblen County	0 916137	0 555135	0 146390	604	6010	93328	0
2016	TN	Hamilton County	0 942223	0 180134	0 212552	448	5010	123557	0
2017	TN	Hancock County	0 994593	0 564740	0 022803	258	2163	60932	0
2018	TN	Hardeman County	0 995736	0 374313	0 006328	240	2149	79314	1
2019	TN	Hardin County	0 969059	0 297991	0 023401	188	2067	78589	1
2020	TN	Hawkins County	0 974032	0 499552	0 086829	302	3920	81692	0
2021	TN	Haywood County	0 998225	0 657081	0 000000	554	2656	87408	0
2022	TN	Henderson County	0 988799	0 441258	0 007743	343	2376	83425	0
2023	TN	Henry County	0 946519	0 532612	0 148927	403	2389	90307	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2024	TN	Hickman County	0 999916	0 331951	0 009174	166	2632	82945	0
2025	TN	Houston County	0 967569	0 346603	0 057883	189	2217	80222	1
2026	TN	Humphreys County	0 955988	0 350576	0 092523	187	3406	87374	0
2027	TN	Jackson County	0 966601	0 441578	0 068864	162	1952	66689	1
2028	TN	Jefferson County	0 871122	0 563020	0 224568	465	5949	90145	0
2029	TN	Johnson County	0 985923	0 285380	0 123340	365	4327	67061	0
2030	TN	Knox County	0 967057	0 289625	0 131858	364	8389	117732	0
2031	TN	Lake County	0 843245	0 873303	0 000000	647	3137	74668	0
2032	TN	Lauderdale County	0 927655	0 606938	0 013172	502	3131	83256	0
2033	TN	Lawrence County	0 998801	0 498035	0 255995	323	3439	89359	0
2034	TN	Lewis County	0 998646	0 204802	0 040149	164	2393	70888	1
2035	TN	Lincoln County	0 999315	0 754052	0 239699	347	2698	93504	0
2036	TN	Loudon County	0 925414	0 503395	0 201992	1293	6750	93687	0
2037	TN	McMinn County	0 995508	0 448661	0 361566	659	3886	91133	0
2038	TN	McNairy County	0 998589	0 339928	0 002056	266	2329	82850	0
2039	TN	Macon County	0 999813	0 705170	0 035224	308	2665	78576	0
2040	TN	Madison County	0 997209	0 396452	0 013694	475	3179	103373	0
2041	TN	Marion County	0 972832	0 158698	0 033287	431	3375	81441	1
2042	TN	Marshall County	0 998018	0 673892	0 0507454	352	3346	97947	0
2043	TN	Maury County	0 995694	0 626307	0 216450	260	3593	113830	0
2044	TN	Meigs County	0 898876	0 451042	0 384402	221	2653	79016	0
2045	TN	Monroe County	0 972988	0 246400	0 483438	580	4501	79734	0
2046	TN	Montgomery County	0 991336	0 506593	0 026096	431	3356	89196	0
2047	TN	Moore County	0 990754	0 581812	0 335016	397	2701	80289	0
2048	TN	Morgan County	0 999344	0 129299	0 117974	298	2455	72060	0
2049	TN	Obion County	0 981218	0 736880	0 012498	499	2719	97649	0
2050	TN	Overton County	0 996628	0 380451	0 232745	248	2442	65307	0
2051	TN	Perry County	0 981079	0 199582	0 000280	167	1731	81190	0
2052	TN	Pickett County	0 933205	0 360121	0 020383	326	2928	58554	0
2053	TN	Polk County	0 983544	0 112645	0 182014	1500	4842	74363	0
2054	TN	Putnam County	0 996337	0 454745	0 206088	282	3896	98550	0
2055	TN	Rhea County	0 939202	0 259446	0 104831	372	2917	93748	0
2056	TN	Roane County	0 913927	0 226947	0 136580	227	3807	104382	0
2057	TN	Robertson County	0 999573	0 765036	0 112109	604	3716	93308	0
2058	TN	Rutherford County	0 991946	0 505113	0 306519	261	4344	104991	0
2059	TN	Scott County	0 997889	0 096060	0 076786	387	3458	75311	1
2060	TN	Sequatchie County	0 999332	0 145939	0 118617	308	3063	75840	0
2061	TN	Sevier County	0 990902	0 195486	0 027423	319	8237	91960	0
2062	TN	Shelby County	0 963229	0 300036	0 010417	483	5499	124830	1
2063	TN	Smith County	0 966385	0 746912	0 093587	243	2558	99742	0
2064	TN	Stewart County	0 927984	0 183646	0 000000	240	3217	78576	0
2065	TN	Sullivan County	0 961219	0 350947	0 171700	594	7815	106658	0
2066	TN	Sumner County	0 974580	0 523996	0 065670	425	5153	111853	0
2067	TN	Tipton County	0 967639	0 623031	0 006365	538	3154	94676	0
2068	TN	Trousdale County	0 979390	0 753516	0 034067	342	3357	87936	0
2069	TN	Unicoi County	0 998120	0 094786	0 063206	340	7361	85904	1
2070	TN	Union County	0 904625	0 345608	0 072845	270	4726	66669	0
2071	TN	Van Buren County	0 995862	0 187929	0 386755	305	2308	66059	0
2072	TN	Warren County	0 996743	0 596937	0 105539	844	3447	89555	0
2073	TN	Washington County	0 989424	0 563303	0 275470	941	7703	108249	0
2074	TN	Wayne County	0 997779	0 266281	0 006539	127	1783	72006	0
2075	TN	Weakley County	0 997295	0 549698	0 104463	560	2232	94635	0
2076	TN	White County	0 992614	0 513415	0 370269	356	3090	86541	0
2077	TN	Williamson County	0 998391	0 548044	0 143323	308	5994	163126	0
2078	TN	Wilson County	0 978346	0 587330	0 135767	220	4143	113512	0
2079	AL	Autauga County	0 985972	0 281190	0 024913	326	1931	102113	1
2080	AL	Baldwin County	0 787602	0 164254	0 027608	704	3825	98909	0
2081	AL	Barbour County	0 978340	0 312836	0 011133	442	2199	88505	1
2082	AL	Bibb County	0 994982	0 120564	0 089310	136	2516	82051	1
2083	AL	Blount County	0 992354	0 332566	0 023420	1743	3983	88966	0
2084	AL	Bullock County	0 998333	0 361960	0 022319	326	1940	66113	0
2085	AL	Butler County	0 998648	0 193925	0 017695	578	2032	73421	0
2086	AL	Calhoun County	0 993721	0 189605	0 026250	1265	3888	93308	0
2087	AL	Chambers County	0 990472	0 286538	0 099783	94	1648	87835	0
2088	AL	Cherokee County	0 921953	0 343203	0 012866	808	2604	85627	0
2089	AL	Chilton County	0 990384	0 223921	0 017596	299	3018	81570	0
2090	AL	Choctaw County	0 992036	0 116216	0 043897	195	2316	77289	1
2091	AL	Clarke County	0 988723	0 077497	0 013906	67	2137	86006	0
2092	AL	Clay County	0 998469	0 176818	0 028893	794	2283	87699	0
2093	AL	Cleburne County	0 998468	0 131648	0 004511	1799	3771	83222	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2094	AL	Coffee County	0 998043	0 403076	0 015210	1334	2513	95211	0
2095	AL	Colbert County	0 953369	0 363009	0 000000	492	2936	90625	0
2096	AL	Conecuh County	0 997990	0 151437	0 084815	199	2820	83534	0
2097	AL	Coosa County	0 979119	0 097779	0 168632	84	2170	72229	1
2098	AL	Covington County	0 991169	0 251410	0 033281	666	2257	86663	0
2099	AL	Crenshaw County	0 997886	0 285308	0 027963	1027	2041	80723	1
2100	AL	Cullman County	0 978299	0 416520	0 012245	2981	4947	93430	0
2101	AL	Dale County	0 997164	0 374683	0 002642	639	2564	92603	0
2102	AL	Dallas County	0 987253	0 371864	0 049779	264	1678	80913	0
2103	AL	DeKalb County	0 999059	0 423236	0 018832	1902	4169	97358	0
2104	AL	Elmore County	0 945710	0 262346	0 152503	423	2681	100717	0
2105	AL	Escambia County	0 994158	0 141613	0 038201	433	2604	82633	0
2106	AL	Etowah County	0 974548	0 250721	0 044011	939	3446	90720	0
2107	AL	Fayette County	0 997505	0 161162	0 204614	251	2279	83663	0
2108	AL	Franklin County	0 983194	0 319693	0 030426	942	2812	93037	0
2109	AL	Geneva County	0 995549	0 530073	0 020900	1015	2501	94296	0
2110	AL	Greene County	0 978746	0 310481	0 056789	173	1801	66086	0
2111	AL	Hale County	0 980595	0 406734	0 096129	379	1794	61264	1
2112	AL	Henry County	0 988700	0 464198	0 011427	609	2176	86961	1
2113	AL	Houston County	0 997769	0 516365	0 042571	729	3104	106651	0
2114	AL	Jackson County	0 957388	0 296171	0 031606	574	2489	86473	0
2115	AL	Jefferson County	0 990038	0 050200	0 073071	334	6159	126144	1
2116	AL	Lamar County	0 998980	0 144918	0 178982	203	2413	85586	0
2117	AL	Lauderdale County	0 931390	0 471166	0 040781	329	2963	97554	0
2118	AL	Lawrence County	0 965629	0 390865	0 023707	803	3009	79673	0
2119	AL	Lee County	0 988916	0 174429	0 010935	391	3324	91235	1
2120	AL	Limestone County	0 935707	0 569964	0 017695	642	4204	99004	0
2121	AL	Lowndes County	0 990224	0 434613	0 020799	284	1636	75413	0
2122	AL	Macon County	0 995568	0 354269	0 162408	163	1966	66553	0
2123	AL	Madison County	0 990248	0 435523	0 034444	533	4420	128623	0
2124	AL	Marengo County	0 994095	0 318405	0 245353	173	1516	83527	0
2125	AL	Marion County	0 997101	0 188031	0 034068	435	2061	74512	0
2126	AL	Marshall County	0 909973	0 393646	0 014054	2346	5343	102980	0
2127	AL	Mobile County	0 750157	0 132187	0 038497	1296	5319	97460	0
2128	AL	Monroe County	0 991639	0 167625	0 030646	322	2241	84191	0
2129	AL	Montgomery County	0 987577	0 457438	0 042283	325	2875	118321	0
2130	AL	Morgan County	0 971885	0 418406	0 096047	913	3925	111473	0
2131	AL	Perry County	0 993654	0 313120	0 167354	229	1480	62632	0
2132	AL	Pickens County	0 990297	0 188264	0 027404	1142	2133	77336	0
2133	AL	Pike County	0 998420	0 417511	0 014453	563	1999	87632	0
2134	AL	Randolph County	0 994757	0 259304	0 020153	802	2339	81746	0
2135	AL	Russell County	0 990268	0 274464	0 065527	158	1736	87191	1
2136	AL	St. Clair County	0 969909	0 192669	0 023563	1261	3952	88431	0
2137	AL	Shelby County	0 981868	0 140928	0 225212	432	5877	107931	0
2138	AL	Sumter County	0 990840	0 289926	0 009984	159	1379	70888	0
2139	AL	Talladega County	0 972742	0 220134	0 083389	352	2922	82714	0
2140	AL	Tallapoosa County	0 937022	0 171667	0 005960	184	1969	96416	0
2141	AL	Tuscaloosa County	0 980205	0 113414	0.100399	528	3129	102363	0
2142	AL	Walker County	0 986458	0 098844	0 003367	1460	3311	98591	0
2143	AL	Washington County	0 992756	0 123015	0 001842	473	2424	84712	0
2144	AL	Wilcox County	0 979303	0 248346	0 032550	126	1539	70922	1
2145	AL	Winston County	0 972433	0 144125	0 048529	2398	3547	81434	0
2146	MS	Adams County	0 946561	0 271503	0 000000	230	1860	91865	0
2147	MS	Alcorn County	0 996466	0 307267	0 051791	203	2135	85959	0
2148	MS	Amite County	0 997208	0 241757	0 603258	238	2230	63885	0
2149	MS	Attala County	0 997331	0 228527	0 026289	206	1404	71796	0
2150	MS	Benton County	0 995451	0 350923	0 004739	360	1523	68971	0
2151	MS	Bolivar County	0 967460	0 760591	0 002123	673	2149	76300	1
2152	MS	Calhoun County	0 997578	0 339231	0 014143	358	1782	74444	0
2153	MS	Carroll County	0 989240	0 377688	0 027829	433	1740	69324	1
2154	MS	Chickasaw County	0 994631	0 464231	0 051434	401	1565	79057	0
2155	MS	Choctaw County	0 998305	0 159229	0 115676	308	1464	65937	1
2156	MS	Claiborne County	0 970839	0 284119	0 015256	225	1594	67075	1
2157	MS	Clarke County	0 996909	0 155190	0 057107	116	1880	71870	0
2158	MS	Clay County	0 981986	0 483231	0 081026	300	1675	83249	0
2159	MS	Coahoma County	0 950288	0 830451	0 000000	746	2712	85640	0
2160	MS	Copiah County	0 996429	0 254721	0 084644	539	1869	70631	0
2161	MS	Covington County	0 997238	0 301920	0 008892	857	3313	67725	1
2162	MS	DeSoto County	0 962677	0 456035	0 049638	428	3014	109523	0
2163	MS	Forrest County	0 992619	0 124358	0 057630	569	3036	89589	1

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2164	MS	Franklin County	0 996219	0 128766	0 239676	137	1813	76686	1
2165	MS	George County	0 988961	0 142096	0 101348	392	2971	71024	0
2166	MS	Greene County	0 991940	0 108008	0 024346	508	2556	53732	1
2167	MS	Grenada County	0 938820	0 369322	0 000000	363	1446	85132	0
2168	MS	Hancock County	0 863096	0 098453	0 481890	186	2926	84143	0
2169	MS	Harrison County	0 595202	0 044816	0 143004	289	6872	92732	0
2170	MS	Hinds County	0 990713	0 414932	0 010469	378	2632	109827	1
2171	MS	Holmes County	0 989295	0 461697	0 000770	431	2226	63953	0
2172	MS	Humphreys County	0 969680	0 673028	0 000000	1251	2552	96477	0
2173	MS	Issaquena County	0 935662	0 430283	0 000000	607	1987	76706	0
2174	MS	Itawamba County	0 985043	0 225030	0 006918	657	1906	77871	0
2175	MS	Jackson County	0 696441	0 053424	0 050282	488	4830	91269	1
2176	MS	Jasper County	0 997870	0 206088	0 067846	429	1617	68592	0
2177	MS	Jefferson County	0 985194	0 199308	0 000000	182	1377	55120	0
2178	MS	Jefferson Davis County	0 998299	0 309493	0 021873	278	2400	59997	0
2179	MS	Jones County	0 991556	0 218244	0 024431	1706	3526	83933	0
2180	MS	Kemper County	0 998851	0 190380	0 055656	117	1481	60850	0
2181	MS	Lafayette County	0 929081	0 244632	0 027515	112	1575	78705	1
2182	MS	Lamar County	0 993784	0 167747	0 038987	620	3578	69940	0
2183	MS	Lauderdale County	0 983545	0 179184	0 062088	158	2130	103142	0
2184	MS	Lawrence County	0 988253	0 227976	0 093809	272	1803	71227	0
2185	MS	Leake County	0 995422	0 256692	0 036591	1348	2921	74614	0
2186	MS	Lee County	0 992259	0 487200	0 139573	353	2136	99072	0
2187	MS	Leflore County	0 976215	0 692517	0 000000	856	2593	90158	0
2188	MS	Lincoln County	0 995806	0 263854	0 467175	380	2198	79626	0
2189	MS	Lowndes County	0 972557	0 391049	0 022634	321	1848	94825	1
2190	MS	Madison County	0 969148	0 432278	0 003039	323	3003	100067	0
2191	MS	Marion County	0 988649	0 258743	0 430401	264	2116	71322	0
2192	MS	Marshall County	0 995119	0 402593	0 125461	214	2474	77445	0
2193	MS	Monroe County	0 989808	0 358267	0 054740	288	1858	83392	0
2194	MS	Montgomery County	0 997511	0 308257	0 008701	248	1446	73245	0
2195	MS	Neshoba County	0 997126	0 376251	0 053337	744	2277	72602	0
2196	MS	Newton County	0 997331	0 260764	0 118482	899	1909	85153	0
2197	MS	Noxubee County	0 992553	0 453675	0 138186	331	1616	60566	0
2198	MS	Oktibbeha County	0 990950	0 275669	0 494844	291	2175	76090	0
2199	MS	Panola County	0 970306	0 498158	0 005509	358	1940	76300	0
2200	MS	Pearl River County	0 990920	0 179404	0 096476	264	2994	76361	0
2201	MS	Perry County	0 995352	0 076257	0 015434	625	2665	69635	1
2202	MS	Pike County	0 995402	0 306988	0 662467	530	2858	77133	0
2203	MS	Pontotoc County	0 992831	0 390139	0 094509	247	1986	86446	0
2204	MS	Prentiss County	0 992111	0 324175	0 119868	236	1868	77546	0
2205	MS	Quitman County	0 995928	0 718975	0 000000	547	2353	71356	0
2206	MS	Rankin County	0 960857	0 239339	0 023550	833	2461	98381	0
2207	MS	Scott County	0 997867	0 282482	0 010949	3031	3320	87991	1
2208	MS	Sharkey County	0 983505	0 664631	0 000000	810	2376	81292	0
2209	MS	Simpson County	0 996957	0.256199	0 005892	2155	3296	74072	0
2210	MS	Smith County	0 997816	0.233717	0 009225	2326	3533	76408	0
2211	MS	Stone County	0 993966	0 114598	0 011479	343	3342	77397	0
2212	MS	Sunflower County	0 981016	0 812975	0 000000	885	2262	83541	0
2213	MS	Tallahatchie County	0 987487	0.662678	0 000420	560	2113	71735	0
2214	MS	Tate County	0 984258	0 545583	0 276967	362	2149	91445	0
2215	MS	Tippah County	0 995510	0 369573	0 165110	227	1862	80046	0
2216	MS	Tishomingo County	0 954084	0.149842	0 127977	117	1965	76070	1
2217	MS	Tunica County	0 945979	0 791906	0 000000	674	2379	77275	0
2218	MS	Union County	0 996673	0 377662	0 062399	275	2003	80987	0
2219	MS	Walton County	0 998469	0 418767	0 688482	482	2637	58636	0
2220	MS	Warren County	0 948038	0 303843	0 000000	343	2144	101869	0
2221	MS	Washington County	0 951043	0 738563	0 000000	821	2632	83554	0
2222	MS	Wayne County	0 996115	0 139816	0 008731	962	2582	69852	1
2223	MS	Webster County	0 998759	0 279173	0 032834	282	1508	77025	0
2224	MS	Wilkinson County	0 984074	0.207348	0 213455	144	1591	65829	0
2225	MS	Winston County	0 994914	0.214783	0 157141	186	1894	77817	0
2226	MS	Yalobusha County	0 943688	0 261657	0 047340	216	1524	80181	0
2227	MS	Yazoo County	0 984327	0 614484	0 008246	511	1969	83121	1
2228	AR	Arkansas County	0 956166	0 650385	0 000000	646	2759	99810	0
2229	AR	Ashley County	0 981044	0 256627	0 001398	738	2360	97168	1
2230	AR	Baxter County	0 944842	0 261276	0 013111	604	2256	99769	0
2231	AR	Benton County	0 961788	0 544255	0 033786	2287	4880	107944	0
2232	AR	Boone County	0 982316	0 652872	0 047089	402	2410	95699	0
2233	AR	Bradley County	0 994267	0 072511	0 000000	978	3262	94655	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2234	AR	Calhoun County	0 993261	0 046796	0 000000	118	2559	65822	0
2235	AR	Carroll County	0 986709	0 606924	0 041181	1102	2448	93179	0
2236	AR	Chicot County	0 932176	0 652885	0 003598	644	2352	69879	1
2237	AR	Clark County	0 980547	0 178579	0 094957	279	1653	87720	0
2238	AR	Clay County	0 996696	0 766339	0 000960	518	2633	86961	1
2239	AR	Cleburne County	0 934350	0 305232	0 068637	820	2569	83906	0
2240	AR	Cleveland County	0 998184	0 089175	0 009390	1948	2549	90659	1
2241	AR	Columbia County	0 999025	0 116757	0 002209	1234	3038	99044	1
2242	AR	Conway County	0 981490	0 470732	0 096750	932	2482	88621	0
2243	AR	Craighead County	0 996869	0 770261	0 001363	653	2973	92298	1
2244	AR	Crawford County	0 985532	0 382411	0 028224	806	3555	83961	0
2245	AR	Crittenden County	0 958795	0 836427	0 000000	596	2322	84164	0
2246	AR	Cross County	0 989583	0 823342	0 000000	548	2612	87666	0
2247	AR	Dallas County	0 998893	0 048197	0 000000	143	2368	88018	0
2248	AR	Desha County	0 933461	0 535147	0 000000	679	2295	83337	0
2249	AR	Drew County	0 991059	0 208008	0 009104	466	2297	82267	1
2250	AR	Faulkner County	0 974872	0 508530	0 304465	223	2359	96897	0
2251	AR	Franklin County	0 983657	0 432548	0 050222	1197	2460	87042	0
2252	AR	Fulton County	0 996571	0 565846	0 300029	155	1292	66946	0
2253	AR	Garland County	0 923013	0 098614	0 037006	477	3521	107924	0
2254	AR	Grant County	0 998053	0 093000	0 033782	368	3068	96227	1
2255	AR	Greene County	0 996313	0 680968	0 012192	498	2667	81990	0
2256	AR	Hempstead County	0 983044	0 361974	0 003924	1712	2615	84557	0
2257	AR	Hot Spring County	0 988387	0 199445	0 086898	302	2454	79727	0
2258	AR	Howard County	0 986943	0 281190	0 004332	2093	3083	106583	0
2259	AR	Independence County	0 989909	0 538354	0 011587	512	1999	89948	0
2260	AR	Izard County	0 994308	0 494775	0 059619	302	1339	89738	0
2261	AR	Jackson County	0 987648	0 907475	0 001933	439	2033	85803	1
2262	AR	Jefferson County	0 968303	0 497758	0 000000	656	2281	93531	0
2263	AR	Johnson County	0 969868	0 256982	0 001277	1404	2714	80337	0
2264	AR	Lafayette County	0 965932	0 320015	0 002729	1060	1634	96234	1
2265	AR	Lawrence County	0 990250	0 750854	0 006091	452	2320	82951	0
2266	AR	Lee County	0 971261	0 775272	0 000000	541	1931	69642	0
2267	AR	Lincoln County	0 980832	0 519735	0 003178	789	2166	63187	1
2268	AR	Little River County	0 941338	0 420486	0 010712	407	1794	95218	1
2269	AR	Logan County	0 970430	0 411202	0 095737	757	2339	86934	0
2270	AR	Lonoke County	0 953916	0 781174	0 035916	678	2664	98503	0
2271	AR	Madison County	0 999798	0 500465	0 039429	769	2566	88749	0
2272	AR	Marion County	0 933402	0 373430	0 057600	284	2057	85417	0
2273	AR	Miller County	0 978896	0 435298	0 005294	502	2264	92522	0
2274	AR	Mississippi County	0 976644	0 843160	0 000000	731	2914	81339	0
2275	AR	Monroe County	0 976289	0 565138	0 000000	523	2365	84082	0
2276	AR	Montgomery County	0 975801	0 159662	0 013793	1001	3026	78874	0
2277	AR	Nevada County	0 998672	0 174952	0 001982	772	1931	82945	0
2278	AR	Newton County	0 999746	0 194708	0 112489	138	1892	66242	0
2279	AR	Ouachita County	0 990290	0 068264	0 000000	400	2518	85207	0
2280	AR	Perry County	0 982966	0 190134	0 006476	956	2430	77912	0
2281	AR	Phillips County	0 952404	0 806185	0 000000	645	1996	77059	0
2282	AR	Pike County	0.982290	0 183627	0 005035	1312	2989	89169	1
2283	AR	Poinsett County	0 992604	0 834217	0 000000	670	2770	83676	0
2284	AR	Polk County	0 996501	0 223381	0 004433	1511	3417	79931	0
2285	AR	Pope County	0 977266	0 300900	0 015099	1474	3797	92271	0
2286	AR	Prairie County	0 955891	0 757621	0 019610	550	2264	80730	0
2287	AR	Pulaski County	0 954257	0 226778	0 005608	370	2794	123442	0
2288	AR	Randolph County	0 993584	0 608699	0 007742	283	2326	73110	0
2289	AR	St Francis County	0 986677	0 752803	0 000585	484	1908	70272	0
2290	AR	Saline County	0 992140	0 098328	0 034268	200	3861	94676	1
2291	AR	Scott County	0 995310	0.200591	0 008750	1181	2553	81102	0
2292	AR	Searcy County	0 997967	0 457860	0 313708	152	1937	73557	0
2293	AR	Sebastian County	0 982231	0 335065	0 070430	588	3503	106143	0
2294	AR	Sevier County	0.970059	0 363915	0 003442	1678	2837	95468	0
2295	AR	Sharp County	0 996715	0 411081	0 023517	374	1721	78190	0
2296	AR	Stone County	0 995346	0 351087	0 006418	565	1957	70895	0
2297	AR	Union County	0 984489	0 046905	0 001604	3118	4866	110579	1
2298	AR	Van Buren County	0 982325	0 263354	0 349105	350	2114	84238	0
2299	AR	Washington County	0 993906	0 579328	0 026474	1995	4468	99810	0
2300	AR	White County	0 992056	0 542269	0 100121	359	2037	80527	0
2301	AR	Woodruff County	0 987392	0 732086	0 000000	472	2017	81339	0
2302	AR	Yell County	0 977871	0 320556	0 010463	1177	2538	85593	0
2303	LA	Acadia Parish	0 996452	0 637841	0 004581	417	2671	74397	1

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2304	LA	Allen Parish	0 998502	0 240327	0 025641	219	2372	66262	0
2305	LA	Ascension Parish	0 962522	0 340017	0 013729	586	6147	90930	1
2306	LA	Assumption Parish	0 929047	0 313350	0 000000	742	2989	77391	0
2307	LA	Avoyelles Parish	0 961605	0 480703	0 011808	481	2679	68267	0
2308	LA	Beauregard Parish	0 994948	0 183883	0 242999	188	2477	83974	0
2309	LA	Bienville Parish	0 986471	0 099066	0 185084	265	2764	81827	0
2310	LA	Bossier Parish	0 967072	0 205176	0 046507	298	2489	94981	1
2311	LA	Caddo Parish	0 941418	0 301748	0 047170	318	2245	110654	1
2312	LA	Calcasieu Parish	0 978786	0 477720	0 000000	117	2318	101381	0
2313	LA	Caldwell Parish	0 979094	0 195895	0 000000	387	2495	79456	0
2314	LA	Cameron Parish	0 679709	0 307059	0 028065	71	2013	85566	1
2315	LA	Catahoula Parish	0 951606	0 556441	0 000000	468	1980	68497	0
2316	LA	Claiborne Parish	0 983185	0 131843	0 113607	720	2219	75792	0
2317	LA	Concordia Parish	0 930075	0 510896	0 000000	489	2011	84773	0
2318	LA	De Soto Parish	0 980673	0 263294	0 694077	312	2284	79931	0
2319	LA	East Baton Rouge Parish	0 968300	0 271407	0 266113	278	6333	121112	0
2320	LA	East Carroll Parish	0 952510	0 719364	0 000000	666	2562	73665	0
2321	LA	East Feliciana Parish	0 994985	0 471440	0 191288	153	2726	82335	0
2322	LA	Evangeline Parish	0 977443	0 416200	0 014152	418	2576	75840	0
2323	LA	Franklin Parish	0 981022	0 662476	0 002750	612	2057	65937	0
2324	LA	Grant Parish	0 970685	0 107756	0 000000	252	2271	68992	0
2325	LA	Iberia Parish	0 557918	0 299291	0 009711	954	5169	83317	1
2326	LA	Iberville Parish	0 947609	0 205581	0 000000	824	3820	87110	0
2327	LA	Jackson Parish	0 982050	0 046839	0 013545	2206	3452	81996	1
2328	LA	Jefferson Parish	0 476207	0 021078	0 000000	850	4932	115828	0
2329	LA	Jefferson Davis Parish	0 990488	0 698250	0 000000	322	2208	75609	0
2330	LA	Lafayette Parish	0 998313	0 507044	0 022436	517	4777	114250	0
2331	LA	Lafourche Parish	0 736817	0 191101	0 007630	504	3328	87300	1
2332	LA	La Salle Parish	0 941830	0 068795	0 000000	123	2569	78528	0
2333	LA	Lincoln Parish	0 998111	0 194664	0 044521	1053	3074	86108	0
2334	LA	Livingston Parish	0 922007	0 086939	0 059805	674	3563	75955	0
2335	LA	Madison Parish	0 959344	0 617192	0 000000	550	1948	62374	0
2336	LA	Morehouse Parish	0 986401	0 475510	0 005692	742	2411	83669	1
2337	LA	Natchitoches Parish	0.966939	0 227089	0 024781	564	2038	72575	1
2338	LA	Orleans Parish	0 515796	0 000865	0 000000	10798	81900	112286	0
2339	LA	Ouachita Parish	0 965600	0 190975	0 011272	607	3055	93937	1
2340	LA	Plaquemines Parish	0 347772	0 085298	0 000000	134	3617	100149	0
2341	LA	Pointe Coupee Parish	0 943554	0 541403	0 000000	558	2887	82728	0
2342	LA	Rapides Parish	0 971090	0 248752	0 019445	531	3017	98990	0
2343	LA	Red River Parish	0 966421	0 392611	0 000000	214	1498	75460	0
2344	LA	Richland Parish	0 989281	0 691318	0 007084	583	2418	82647	1
2345	LA	Sabine Parish	0 855426	0 104346	0 028732	1603	3218	78989	0
2346	LA	St Bernard Parish	0 259312	0 020712	0 000000	180	5209	93023	0
2347	LA	St Charles Parish	0 691539	0 127712	0 000000	432	3090	109502	0
2348	LA	St Helena Parish	0 997326	0 192519	0 460822	1113	3146	59584	0
2349	LA	St James Parish	0 954767	0 272460	0 000000	1124	4874	100088	0
2350	LA	St John the Baptist Parish	0 629306	0 123815	0 000000	635	3635	90402	0
2351	LA	St Landry Parish	0 989158	0 476363	0.013262	416	2593	76273	0
2352	LA	St Martin Parish	0 906179	0 149799	0 008336	689	3595	73184	0
2353	LA	St Mary Parish	0 547788	0 208390	0 000000	803	4057	84177	0
2354	LA	St Tammany Parish	0 760092	0 073478	0 124129	679	6640	116018	0
2355	LA	Tangipahoa Parish	0 960062	0 250863	0 715459	1200	4382	79274	0
2356	LA	Tensas Parish	0 939619	0 637895	0 003573	581	2085	81610	1
2357	LA	Terrebonne Parish	0 603373	0 054960	0 000000	464	3418	85566	0
2358	LA	Union Parish	0 969428	0 110168	0 026775	2383	3302	83974	0
2359	LA	Vermilion Parish	0 763034	0 421539	0 005909	357	2892	79870	0
2360	LA	Vernon Parish	0 990236	0 060922	0 193255	498	3099	65043	0
2361	LA	Washington Parish	0 990532	0 271196	0 736909	823	3197	77099	0
2362	LA	Webster Parish	0 968788	0 154145	0 229665	149	2389	88959	0
2363	LA	West Baton Rouge Parish	0 938998	0 315128	0 000000	759	4117	94567	0
2364	LA	West Carroll Parish	0 997442	0 551394	0 020784	581	1962	64921	1
2365	LA	West Feliciana Parish	0 953037	0 334237	0 060502	194	2719	66330	1
2366	LA	Winn Parish	0 993281	0 037953	0 000000	315	2338	68897	0
2367	OK	Adair County	0 997643	0 562717	0 094086	1053	2073	74309	0
2368	OK	Alfalfa County	0 983216	0 879113	0 006104	467	1860	129388	0
2369	OK	Atoka County	0 988194	0 599986	0 039436	117	1002	60200	0
2370	OK	Beaver County	0 998227	0 849616	0 016307	199	785	131773	0
2371	OK	Beckham County	0 997447	0 855212	0 063305	129	1137	88593	0
2372	OK	Blaine County	0 988963	0 864545	0 008407	342	1693	103847	0
2373	OK	Bryan County	0 963290	0 709469	0 060914	204	1440	77750	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2374	OK	Caddo County	0 990717	0 887908	0 007909	302	1689	89474	0
2375	OK	Canadian County	0 994089	0 867940	0 053348	356	2209	105662	0
2376	OK	Carter County	0 988094	0 707235	0 004543	132	1317	105039	1
2377	OK	Cherokee County	0 967331	0 455179	0 049212	878	1901	77323	0
2378	OK	Choctaw County	0 966590	0 607317	0 030152	185	1155	73408	0
2379	OK	Cimarron County	0 996671	0 881214	0 000512	424	568	166661	1
2380	OK	Cleveland County	0 960199	0 457833	0 137787	199	2732	97825	0
2381	OK	Coal County	0 994102	0 798632	0 105639	124	1034	69588	0
2382	OK	Comanche County	0 986641	0 592942	0 162405	187	1480	89596	0
2383	OK	Cotton County	0 991738	0 879662	0 003009	208	1271	99431	1
2384	OK	Craig County	0 997792	0 918111	0 031580	298	1197	97128	0
2385	OK	Creek County	0 985332	0 549862	0 147087	91	1382	98733	0
2386	OK	Custer County	0 984541	1 003900	0 016665	254	1534	98076	9
2387	OK	Delaware County	0 934848	0 510667	0 072922	752	2055	72995	0
2388	OK	Dewey County	0 991936	0 910895	0 016520	149	1056	111534	0
2389	OK	Ellis County	0 997808	0 889163	0 057426	119	762	115050	0
2390	OK	Garfield County	0 998584	0 977384	0.013820	288	1813	112889	0
2391	OK	Garvin County	0 994391	0 810561	0 050400	190	1570	97791	0
2392	OK	Grady County	0 996081	0 803425	0 342118	345	1642	87469	0
2393	OK	Grant County	0 996964	0 936189	0 012319	224	1507	146816	0
2394	OK	Greer County	0 993312	0 827251	0 002514	132	1000	88438	1
2395	OK	Harmon County	0 998632	0 831934	0 004550	172	931	103704	1
2396	OK	Harper County	0 998117	0 913051	0 009757	359	811	132612	0
2397	OK	Haskell County	0 922863	0 725762	0 007878	165	1330	80940	0
2398	OK	Hughes County	0 990298	0 671202	0 017688	112	1055	77011	0
2399	OK	Jackson County	0 998169	0 914596	0 001996	263	1405	90734	0
2400	OK	Jefferson County	0 980592	0 834453	0 001366	183	1082	88891	0
2401	OK	Johnston County	0 979072	0 784301	0 056588	207	1169	70211	0
2402	OK	Kay County	0 972061	0 812298	0 018111	243	1777	118172	0
2403	OK	Kingfisher County	0 996733	0 902861	0 044774	361	1813	109252	0
2404	OK	Kiowa County	0 984420	0 859741	0 004949	203	1307	105174	1
2405	OK	Latimer County	0 990471	0 420264	0 000000	112	1272	73584	0
2406	OK	Le Flore County	0 986251	0 374883	0 008665	431	1916	82233	0
2407	OK	Lincoln County	0 992713	0 648553	0 224954	129	1527	81983	0
2408	OK	Logan County	0 994202	0 722420	0 035062	260	1799	90761	0
2409	OK	Love County	0 968865	0 774649	0 004715	186	1472	88986	1
2410	OK	McClain County	0 981920	0 695700	0 149984	309	2072	95387	0
2411	OK	McCurtain County	0 974195	0 265694	0 001257	718	1936	75724	0
2412	OK	McIntosh County	0 870212	0 596668	0 105869	132	1418	74248	0
2413	OK	Major County	0 998846	0 807165	0 038070	233	1453	105770	0
2414	OK	Marshall County	0 869207	0 684929	0 000000	145	1766	87869	0
2415	OK	Mayes County	0 959959	0 661927	0 310805	245	2039	86020	0
2416	OK	Murray County	0 984407	0 862194	0 156216	205	1231	77878	0
2417	OK	Muskogee County	0 970035	0 667106	0 085371	264	1788	91784	0
2418	OK	Noble County	0 985811	0 834565	0 015426	209	1354	99654	0
2419	OK	Nowata County	0 972563	0 781702	0 053489	231	1152	86026	0
2420	OK	Oklfuskee County	0 993393	0 644162	0 009567	118	1054	71843	0
2421	OK	Oklahoma County	0 987153	0 345367	0 054477	171	2986	120163	0
2422	OK	Okmulgee County	0 992394	0 628869	0 019806	144	1503	81150	0
2423	OK	Osage County	0 976999	0 774735	0 012996	183	854	78820	0
2424	OK	Ottawa County	0 972324	0 686583	0 072513	505	2088	82877	0
2425	OK	Pawnee County	0 957254	0 774298	0 002374	306	1213	94710	0
2426	OK	Payne County	0 984522	0 748881	0 220820	201	1748	94344	0
2427	OK	Pittsburg County	0 947816	0 575752	0 038476	127	1091	83541	0
2428	OK	Pontotoc County	0 992015	0 766467	0 080141	207	1368	90104	0
2429	OK	Pottawatomie County	0 993061	0 593484	0 124968	130	1695	92095	0
2430	OK	Pushmataha County	0 982100	0 268324	0 018209	105	1099	66201	0
2431	OK	Roger Mills County	0 996002	0 903357	0 106380	107	975	102174	0
2432	OK	Rogers County	0 948712	0 716701	0 132203	184	2460	100555	0
2433	OK	Seminole County	0 987467	0 619932	0 065244	95	1240	82261	0
2434	OK	Sequoyah County	0 942581	0 499030	0 026002	131	1642	72676	0
2435	OK	Stephens County	0 984244	0 750357	0 055140	132	1169	94154	0
2436	OK	Texas County	0 994312	0 806349	0 000393	1185	977	130431	0
2437	OK	Tillman County	0 992151	0 861410	0 003368	208	1233	97006	1
2438	OK	Tulsa County	0 971523	0 367375	0 013408	252	3899	130689	0
2439	OK	Wagoner County	0 952708	0 601172	0 091730	227	2347	99329	0
2440	OK	Washington County	0 982717	0 810624	0 004049	206	1795	135213	0
2441	OK	Washita County	0 994402	0 899509	0 014415	298	1601	99796	0
2442	OK	Woods County	0 997294	0 902979	0 001409	226	1166	126591	0
2443	OK	Woodward County	0 997021	0 864391	0 001357	120	890	101205	1

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2444	TX	Anderson County	0 993400	0 514284	0 086027	182	2363	84800	0
2445	TX	Andrews County	0 999760	1 002194	0 015915	26	317	103738	13
2446	TX	Angelina County	0 927255	0 200887	0 010473	424	3756	102174	0
2447	TX	Aransas County	0 477181	0 118641	0 000000	46	2960	95956	0
2448	TX	Archer County	0 982624	1 052326	0 349618	212	874	110559	9
2449	TX	Armstrong County	0 999806	0 856201	0 000000	153	548	147249	0
2450	TX	Atascosa County	0 997156	0 970231	0 065235	176	1528	78948	0
2451	TX	Austin County	0 994240	0 807641	0 002805	256	4261	122730	0
2452	TX	Bailey County	0 999172	0 818225	0 013679	399	984	133148	1
2453	TX	Bandera County	0 992689	0 782479	0 067219	30	1821	105824	1
2454	TX	Bastrop County	0 991588	0 694542	0 012385	143	3036	89677	0
2455	TX	Baylor County	0 966437	0 642223	0 000000	204	815	116526	0
2456	TX	Bee County	0 999807	0.784908	0 041026	109	1558	85038	1
2457	TX	Bell County	0 974060	0 614724	0 011326	202	2849	91682	0
2458	TX	Bexar County	0 992176	0 512147	0 038881	274	3903	105039	0
2459	TX	Blanco County	0 996966	0 815537	0 032561	98	2127	120902	1
2460	TX	Borden County	0 992004	1 094580	0 006847	91	430	168632	13
2461	TX	Bosque County	0.986596	0.865261	0 039399	174	1920	99051	0
2462	TX	Bowie County	0 962193	0 462934	0 121977	340	2082	102621	0
2463	TX	Brazoria County	0 868137	0 635408	0 007360	190	2713	121586	1
2464	TX	Brazos County	0 992345	0 788418	0 023845	278	2723	85620	0
2465	TX	Brewster County	0 999971	0 606784	0 026220	12	277	82105	1
2466	TX	Briscoe County	0.998521	0 709525	0 021363	90	527	135030	1
2467	TX	Brooks County	0 999653	0 938151	0 052148	83	1172	67651	0
2468	TX	Brown County	0 986374	0 850005	0 195161	153	1323	93998	0
2469	TX	Burleson County	0 981974	0 744580	0 009105	241	2633	89115	0
2470	TX	Burnet County	0 975609	0 860850	0 008010	59	2455	109137	0
2471	TX	Caldwell County	0 996946	0 755574	0 004643	288	2380	86643	0
2472	TX	Calhoun County	0 496356	0.634541	0 022508	168	1670	104422	1
2473	TX	Callahan County	0 997072	0 853213	0 026161	102	1353	87991	0
2474	TX	Cameron County	0 709458	0 568165	0 000170	617	3281	65890	0
2475	TX	Camp County	0 972015	0 541445	0 032682	4813	2908	122277	0
2476	TX	Carson County	0 999018	1 052898	0 006915	275	1042	121159	13
2477	TX	Cass County	0.976113	0 278242	0 006701	276	2428	92637	0
2478	TX	Castro County	0.998929	0.901434	0 002417	2343	1445	117698	0
2479	TX	Chambers County	0 690041	0 654989	0 000000	147	1425	89975	0
2480	TX	Cherokee County	0 990863	0 399632	0 296883	769	3187	89664	0
2481	TX	Childress County	0 995414	0 989719	0 000000	108	676	97615	0
2482	TX	Clay County	0 983553	0 954200	0 221002	169	1081	101327	0
2483	TX	Cochran County	0 999887	0 746680	0 000000	300	858	116729	0
2484	TX	Coke County	0 968570	0 909183	0 000000	51	820	101618	0
2485	TX	Coleman County	0 993283	0 835375	0 014896	70	931	99248	0
2486	TX	Collin County	0 956883	0 508054	0 084832	250	4610	151063	0
2487	TX	Collingsworth County	0 999301	0 786543	0 000888	144	891	93985	0
2488	TX	Colorado County	0 989068	0 891011	0.012404	202	2258	109435	0
2489	TX	Comal County	0 977119	0 577001	0 002626	59	3408	120854	0
2490	TX	Comanche County	0 989470	0 905995	0 352257	463	1699	98591	0
2491	TX	Concho County	0 997755	0 900892	0 002962	74	1070	105079	0
2492	TX	Cooke County	0 972099	0 769593	0 249629	224	2459	93795	0
2493	TX	Coryell County	0 995307	0 899085	0.018095	108	1612	82091	1
2494	TX	Cottle County	0 999538	0 817441	0 013934	60	679	125128	1
2495	TX	Crane County	0 999963	0 785212	0 000000	18	290	92603	0
2496	TX	Crockett County	0 999996	1 113672	0 000000	19	323	106170	9
2497	TX	Crosby County	0 997588	0 784378	0.004496	194	1599	108114	1
2498	TX	Culberson County	0 999936	0 649297	0 011585	8	267	75609	0
2499	TX	Dallam County	0 999625	0 810867	0 003191	895	991	133534	0
2500	TX	Dallas County	0 968107	0 219766	0 059472	312	4336	142244	0
2501	TX	Dawson County	0 999934	0 957911	0 000000	219	1182	98767	0
2502	TX	Deaf Smith County	0 999383	0 893929	0.000494	1860	990	117556	1
2503	TX	Delta County	0 997243	0 608375	0 191185	196	2029	93165	0
2504	TX	Denton County	0 927716	0 642976	0.019462	296	4583	120570	0
2505	TX	DeWitt County	0 998602	0 978130	0 115277	135	1701	93646	0
2506	TX	Dickens County	0 998891	0 970269	0 026477	53	395	110552	1
2507	TX	Dimmit County	0 997323	0 795108	0 011389	51	1082	55628	1
2508	TX	Donley County	0 996490	1 007659	0 002929	224	742	135782	13
2509	TX	Duval County	0 998347	0 698222	0 120636	55	1077	80818	0
2510	TX	Eastland County	0 993681	0 832174	0 015052	148	1383	87042	0
2511	TX	Ector County	0 999306	0 899543	0 000000	23	307	100013	0
2512	TX	Edwards County	0 999909	0 823381	0 000000	21	566	86751	0
2513	TX	Ellis County	0 987658	0 708437	0 098998	181	3454	107951	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2514	TX	EI Paso County	0 998366	0 999999	0 272530	336	1393	78197	16
2515	TX	Erath County	0 996816	0 836725	0 737455	788	2277	101754	0
2516	TX	Falls County	0 993906	0 767908	0 004902	369	1566	81962	0
2517	TX	Fannin County	0 991475	0 723157	0 045210	166	1710	92644	0
2518	TX	Fayette County	0 989797	0 816901	0 047288	385	3186	114372	0
2519	TX	Fisher County	0 999355	0 946029	0 016810	123	1013	101923	0
2520	TX	Floyd County	0 999732	0 990664	0 001902	432	1121	101043	0
2521	TX	Foard County	0 998570	0 711362	0 017877	68	643	123245	1
2522	TX	Fort Bend County	0 987509	0 754366	0 000401	350	4018	123956	0
2523	TX	Franklin County	0 969080	0 703002	0 339361	890	2172	98130	0
2524	TX	Freestone County	0 992212	0 653310	0 002798	124	2166	91276	0
2525	TX	Frio County	0 998888	1 033514	0 004323	243	1361	68877	13
2526	TX	Gaines County	0 999674	0 714035	0 002373	483	1446	86927	1
2527	TX	Galveston County	0 454871	0 400659	0 040751	189	3918	120184	1
2528	TX	Garza County	0 999302	1 001096	0 006103	89	551	83602	9
2529	TX	Gillespie County	0 999605	1 013528	0 031475	132	2289	117352	9
2530	TX	Glasscock County	0 999795	0 828265	0 014283	115	909	124248	1
2531	TX	Goliad County	0 993197	0 851859	0 010276	72	1504	99437	0
2532	TX	Gonzales County	0 998074	0 973653	0 003759	708	2283	99160	0
2533	TX	Gray County	0 998954	0 969510	0 001974	346	661	123543	1
2534	TX	Grayson County	0 953465	0 685295	0 081455	208	3199	110281	0
2535	TX	Gregg County	0 991524	0 269072	0 120801	207	3046	113620	1
2536	TX	Grimes County	0 990703	0 694656	0 215306	175	2944	86216	0
2537	TX	Guadalupe County	0 995762	0 763038	0 074511	197	3399	93728	0
2538	TX	Hale County	0 999891	0 871415	0 001797	782	1782	103881	1
2539	TX	Hall County	0 998904	0 766469	0 000000	117	544	109319	0
2540	TX	Hamilton County	0 999195	0 862317	0 403261	208	1574	108148	0
2541	TX	Hansford County	0 999348	0 979186	0 000000	1298	979	187482	0
2542	TX	Hardeman County	0 997671	0 705394	0 000000	94	692	111656	0
2543	TX	Hardin County	0 996604	0 056666	0 000000	207	2706	91005	0
2544	TX	Harris County	0 972494	0 278653	0 055896	353	4676	132193	0
2545	TX	Harrison County	0 982157	0 351065	0 042372	142	3378	93646	0
2546	TX	Hartley County	0 999352	0 733596	0 000000	991	835	156359	0
2547	TX	Haskell County	0 991196	0 855054	0 008614	185	1082	111995	1
2548	TX	Hays County	0 997182	1 068156	0 018149	76	2262	93382	9
2549	TX	Hemphill County	0 997390	0 937204	0 003879	372	587	136385	1
2550	TX	Henderson County	0 921276	0 636470	0 096081	251	2971	85762	0
2551	TX	Hidalgo County	0 991139	0 657645	0 006309	756	3404	60275	1
2552	TX	Hill County	0 976385	0 763196	0 169498	234	2123	92664	0
2553	TX	Hockley County	0 999706	0 843910	0 000000	300	1269	92393	0
2554	TX	Hood County	0 965236	0 836961	0 050969	237	3084	129361	0
2555	TX	Hopkins County	0 989856	0 769639	0 752508	877	2645	99072	0
2556	TX	Houston County	0 995202	0 529535	0 028890	172	1928	95746	0
2557	TX	Howard County	0 998501	0 846902	0 016646	97	832	108486	1
2558	TX	Hudspeth County	0 999797	0 763679	0 000000	21	369	88438	0
2559	TX	Hunt County	0 953682	0 641056	0 093066	171	2612	104680	0
2560	TX	Hutchinson County	0 9911530	0 709384	0 000000	571	778	119269	0
2561	TX	Irion County	0 999893	0 976161	0 000000	27	514	142867	0
2562	TX	Jack County	0 997015	0 883990	0 029442	69	1110	91533	0
2563	TX	Jackson County	0 967827	0 869910	0 011027	232	1819	106888	1
2564	TX	Jasper County	0 966779	0 116945	0 000000	131	3116	91086	0
2565	TX	Jeff Davis County	0 999922	1 051950	0 031217	17	243	87171	13
2566	TX	Jefferson County	0 813085	0 557363	0 007366	166	1705	117976	1
2567	TX	Jim Hogg County	0 999952	1 012719	0 000000	24	703	82985	9
2568	TX	Jim Wells County	0 995797	0 935476	0 134595	205	1918	78820	0
2569	TX	Johnson County	0 993113	0 707310	0 560170	385	3926	105574	0
2570	TX	Jones County	0 993459	0 861596	0 005218	169	1137	97148	0
2571	TX	Karnes County	0 995664	0 798772	0 069544	114	1574	82531	0
2572	TX	Kaufman County	0 974289	0 769179	0 059763	166	2597	108121	0
2573	TX	Kendall County	0 999097	0 837071	0 157822	61	2432	150846	0
2574	TX	Kenedy County	0 748800	0 593339	0 000000	41	646	144906	0
2575	TX	Kent County	0 999362	1 030967	0 000000	45	498	107389	9
2576	TX	Kerr County	0 998658	0 750273	0 074833	40	1742	125995	1
2577	TX	Kimble County	0 999820	0 967890	0 012236	25	915	110830	0
2578	TX	King County	0 998865	0 746769	0 000000	44	746	157396	0
2579	TX	Kinney County	0 998633	0 800805	0 020311	28	829	75101	1
2580	TX	Kleberg County	0 798768	0 913932	0 000333	204	1475	93145	17
2581	TX	Knox County	0 998424	1 055310	0 010191	134	654	102214	13
2582	TX	Lamar County	0 983415	0 723610	0 180909	202	1683	100033	0
2583	TX	Lamb County	0 998508	0 796449	0 000000	985	1565	128176	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2584	TX	Lampasas County	0.997308	0.948736	0.056658	70	1675	94175	0
2585	TX	La Salle County	0.996401	0.803857	0.013414	38	877	62354	1
2586	TX	Lavaca County	0.999539	0.850262	0.064430	178	2322	108581	0
2587	TX	Lee County	0.991311	0.792112	0.015681	193	2788	95841	0
2588	TX	Leon County	0.992283	0.702698	0.022342	122	2348	99776	0
2589	TX	Liberty County	0.985971	0.461024	0.004077	156	2027	95597	0
2590	TX	Limestone County	0.973919	0.735909	0.024874	171	1670	88648	0
2591	TX	Lipscomb County	0.999880	0.986444	0.004003	167	519	124214	1
2592	TX	Live Oak County	0.960576	0.842087	0.013352	72	1261	93355	0
2593	TX	Llano County	0.967511	0.847610	0.027747	56	1708	124539	1
2594	TX	Loving County	0.994433	0.804683	0.000000	7	466	170658	0
2595	TX	Lubbock County	0.998736	0.836378	0.000278	684	2063	104599	0
2596	TX	Lynn County	0.998237	0.860171	0.006435	249	1178	114806	1
2597	TX	McCulloch County	0.996230	0.987615	0.007466	78	1156	90829	1
2598	TX	McLennan County	0.982698	0.708322	0.124392	358	2301	101090	0
2599	TX	McMullen County	0.974090	0.724273	0.000000	48	967	188864	0
2600	TX	Madison County	0.994082	0.811678	0.015059	432	2685	95509	0
2601	TX	Marion County	0.906870	0.203199	0.000000	79	2521	77790	0
2602	TX	Martin County	0.999085	0.875261	0.000000	146	1134	122405	0
2603	TX	Mason County	0.999879	0.917618	0.000000	101	1390	112239	0
2604	TX	Matagorda County	0.691267	0.788732	0.000000	196	1647	108066	0
2605	TX	Maverick County	0.990970	0.828245	0.000000	135	998	51686	0
2606	TX	Medina County	0.994932	0.774507	0.024221	197	2155	88749	0
2607	TX	Menard County	0.999625	0.844618	0.010829	75	859	113377	1
2608	TX	Midland County	0.998099	1.257725	0.027580	59	548	131028	13
2609	TX	Milam County	0.995153	0.846949	0.034308	208	1990	96078	0
2610	TX	Mills County	0.997631	0.894349	0.046172	106	1323	109062	0
2611	TX	Mitchell County	0.993605	1.008301	0.000980	90	522	92474	9
2612	TX	Montague County	0.991714	0.831612	0.086989	134	1566	91100	0
2613	TX	Montgomery County	0.969745	0.290087	0.035976	113	3546	115713	0
2614	TX	Moore County	0.989060	1.019864	0.001202	1116	1148	116519	13
2615	TX	Morris County	0.984037	0.453957	0.019670	542	1847	91845	1
2616	TX	Motley County	0.999565	0.757829	0.000000	69	556	121545	0
2617	TX	Nacogdoches County	0.964782	0.363633	0.059415	1414	3310	91303	0
2618	TX	Navarro County	0.986131	0.767079	0.023739	155	1807	95319	0
2619	TX	Newton County	0.992745	0.050703	0.000000	99	3428	67746	0
2620	TX	Nolan County	0.997872	0.918760	0.010815	204	1021	96660	1
2621	TX	Nueces County	0.716533	0.828502	0.003470	256	2395	102438	1
2622	TX	Ochiltree County	0.999440	1.011131	0.000000	515	1000	120814	9
2623	TX	Oldham County	0.999476	0.882488	0.001954	238	425	188512	1
2624	TX	Orange County	0.939035	0.249771	0.054656	126	2382	101388	1
2625	TX	Palo Pinto County	0.966952	0.848109	0.048183	63	1425	94019	0
2626	TX	Panola County	0.975141	0.380682	0.102168	415	2036	105770	0
2627	TX	Parker County	0.992777	0.718836	0.238636	248	3527	112198	0
2628	TX	Parmer County	0.996040	0.932612	0.005841	2059	1690	175426	0
2629	TX	Pecos County	0.999775	0.948395	0.011267	24	250	73855	1
2630	TX	Polk County	0.952658	0.208678	0.037099	105	2300	86738	0
2631	TX	Potter County	0.986324	0.690687	0.000000	135	562	110119	0
2632	TX	Presidio County	0.999806	0.687068	0.011662	20	415	62036	1
2633	TX	Rains County	0.896394	0.662850	0.486137	375	2498	80906	0
2634	TX	Randall County	0.991302	0.849382	0.008443	888	967	120035	0
2635	TX	Reagan County	0.999420	0.821327	0.000000	49	551	88302	0
2636	TX	Real County	0.999814	0.809517	0.000000	29	1347	94452	0
2637	TX	Red River County	0.992976	0.632711	0.047457	217	1480	86622	0
2638	TX	Reeves County	0.997701	0.922251	0.005853	130	319	72487	1
2639	TX	Refugio County	0.940899	1.353270	0.015115	78	1209	118050	13
2640	TX	Roberts County	0.999894	0.862412	0.000000	97	485	135660	0
2641	TX	Robertson County	0.987174	0.716395	0.021062	197	2258	95123	0
2642	TX	Rockwall County	0.866084	0.575598	0.114003	146	4315	144865	1
2643	TX	Runnels County	0.997432	0.848453	0.102509	153	1311	112665	0
2644	TX	Rusk County	0.983960	0.453465	0.101678	199	2250	97168	0
2645	TX	Sabine County	0.850257	0.106899	0.025835	454	2772	84313	1
2646	TX	San Augustine County	0.891366	0.161544	0.002120	683	2428	85586	0
2647	TX	San Jacinto County	0.908830	0.226477	0.114604	125	2026	65876	1
2648	TX	San Patricio County	0.978299	0.809093	0.005652	389	2289	89880	1
2649	TX	San Saba County	0.996678	1.024138	0.006033	116	1178	105967	9
2650	TX	Schleicher County	0.999967	0.911636	0.011810	44	551	92163	1
2651	TX	Scurry County	0.994452	0.897383	0.030957	98	959	98171	1
2652	TX	Shackelford County	0.998268	0.964792	0.059984	58	691	130858	1
2653	TX	Shelby County	0.951590	0.369334	0.005998	1505	3302	92244	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2654	TX	Sherman County	0 999823	0 858939	0 000735	1056	936	211100	1
2655	TX	Smith County	0 977828	0 416713	0 061123	336	3984	118605	0
2656	TX	Somervell County	0 975349	0 524639	0 067791	92	2758	97094	1
2657	TX	Starr County	0 994912	0 808151	0 003384	275	1183	37652	0
2658	TX	Stephens County	0 970881	0 936941	0 028784	51	1118	92393	1
2659	TX	Sterling County	0 999857	1 413412	0 000000	25	521	102106	9
2660	TX	Stonewall County	0 998302	0 871172	0 024848	62	614	132443	1
2661	TX	Sutton County	0 999565	0 995278	0 000000	30	525	106116	0
2662	TX	Swisher County	0 999724	0 876497	0 000759	1185	1196	143626	0
2663	TX	Tarrant County	0 962070	0 303211	0 114371	320	6002	127505	0
2664	TX	Taylor County	0 996044	0 868554	0 006244	314	1469	108513	0
2665	TX	Terrell County	0 999986	0 925261	0 000000	10	270	133114	0
2666	TX	Terry County	0 998812	0 806133	0 005998	286	1266	98536	1
2667	TX	Throckmorton County	0 996579	0 995835	0 023773	85	660	139731	1
2668	TX	Titus County	0 964431	0 686272	0 070508	421	2735	104524	0
2669	TX	Tom Green County	0 988033	1 047782	0 037251	190	1251	107220	9
2670	TX	Travis County	0 967930	0 525607	0 036092	142	3701	124220	0
2671	TX	Trinity County	0 970367	0 247233	0 062484	172	2580	82565	1
2672	TX	Tyler County	0 986321	0 099356	0 003805	133	4090	97521	0
2673	TX	Upshur County	0 991567	0 515780	0 385342	530	2531	87848	0
2674	TX	Upton County	0 999873	0 874393	0 000000	30	320	99742	0
2675	TX	Uvalde County	0 998680	0 920633	0 010327	166	1139	87144	1
2676	TX	Val Verde County	0 980814	0 890307	0 000000	19	419	69940	0
2677	TX	Van Zandt County	0 987497	0 696225	0 194882	340	3619	97846	0
2678	TX	Victoria County	0 992994	0 762560	0 037966	144	1873	112930	1
2679	TX	Walker County	0 982548	0 424441	0 018984	164	2914	87273	0
2680	TX	Waller County	0 990608	0 738930	0 005416	280	5140	95929	0
2681	TX	Ward County	0 999699	0 852490	0 000000	11	313	98171	0
2682	TX	Washington County	0 980490	0 842101	0 108535	219	4330	128278	0
2683	TX	Webb County	0 994460	0 796869	0 002604	33	1050	60776	0
2684	TX	Wharton County	0 996070	0 924033	0 002810	435	2229	103183	1
2685	TX	Wheeler County	0 998779	0 857397	0 002378	330	582	126584	1
2686	TX	Wichita County	0 991555	0 766152	0 020960	162	1501	115950	0
2687	TX	Wilbarger County	0 992809	1 389143	0 009793	93	792	102912	13
2688	TX	Willacy County	0 760854	0 683145	0 000000	414	2468	59895	0
2689	TX	Williamson County	0 989331	0 758296	0 041806	203	2822	101151	0
2690	TX	Wilson County	0 998196	0 922369	0 147676	352	2365	86460	0
2691	TX	Winkler County	0 999770	0 804167	0 000000	19	241	86467	0
2692	TX	Wise County	0 980311	0 796438	0 363915	211	2441	91431	0
2693	TX	Wood County	0 934505	0 489363	0 428419	636	3170	94994	0
2694	TX	Yoakum County	0 999984	0 673362	0 000000	261	930	117752	0
2695	TX	Young County	0 990855	0 954015	0 007073	99	890	120658	1
2696	TX	Zapata County	0 942036	0 760067	0 001639	59	1213	51334	0
2697	TX	Zavala County	0 997513	0 869963	0 012813	128	874	51490	1
2698	MT	Beaverhead County	0 994664	0 378455	0 001241	120	799	101374	0
2699	MT	Big Horn County	0 996043	0 939204	0 000682	76	617	82267	0
2700	MT	Blaine County	0 996995	0 864710	0 003145	45	456	86074	1
2701	MT	Broadwater County	0 961698	0 590085	0 012241	90	835	86521	1
2702	MT	Carbon County	0 993117	0 456752	0 008990	180	1098	97094	0
2703	MT	Carter County	0 997393	0 757689	0 000000	35	243	100623	0
2704	MT	Cascade County	0 994942	0 824811	0 040188	92	742	114907	0
2705	MT	Chouteau County	0 994072	0 895767	0 000682	62	809	115693	0
2706	MT	Custer County	0 997341	0 861181	0 000502	52	372	109577	0
2707	MT	Daniels County	0 999698	0 834267	0 000000	75	534	121362	0
2708	MT	Dawson County	0 995820	0 878300	0 018009	63	409	99248	0
2709	MT	Deer Lodge County	0 994150	0 286502	0 014289	86	926	82626	1
2710	MT	Fallon County	0 998343	0 910765	0 010680	49	354	114372	1
2711	MT	Fergus County	0 997426	0 803904	0 008546	66	528	99173	0
2712	MT	Flathead County	0 969942	0 084904	0 124783	236	3583	104571	0
2713	MT	Gallatin County	0 989782	0 435935	0 178665	206	1861	99234	0
2714	MT	Garfield County	0 962975	0 669515	0 005355	31	287	120157	1
2715	MT	Glacier County	0 986048	0 902901	0 019266	51	549	79490	0
2716	MT	Golden Valley County	0 999068	0 846192	0 011981	43	475	114203	1
2717	MT	Granite County	0 996741	0 316510	0 006634	71	799	106312	1
2718	MT	Hill County	0 993249	0 886882	0 008451	62	681	93687	0
2719	MT	Jefferson County	0 998678	0 346587	0 036954	69	809	110898	0
2720	MT	Judith Basin County	0 999539	0 725364	0 002107	90	623	100629	1
2721	MT	Lake County	0 903289	0 660406	0 107597	144	1632	86101	0
2722	MT	Lewis and Clark County	0 989537	0 398851	0 021675	53	693	111487	0
2723	MT	Liberty County	0 987994	1 040113	0 022465	80	653	132728	9

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2724	MT	Lincoln County	0 983014	0 021720	0 005326	111	3951	85566	0
2725	MT	McCone County	0 984999	0 762822	0 004363	59	472	100006	1
2726	MT	Madison County	0 995476	0 553775	0 000000	71	821	89332	0
2727	MT	Meagher County	0 998739	0 595873	0 003639	50	481	96674	1
2728	MT	Mineral County	0 997093	0 024539	0 000000	125	2698	80452	0
2729	MT	Missoula County	0 992294	0 149270	0 017280	77	1953	105357	1
2730	MT	Musselshell County	0 997988	0 863495	0 014223	34	411	100345	1
2731	MT	Park County	0 996037	0 457546	0 019650	63	933	89447	23
2732	MT	Petroleum County	0 987986	0 632503	0 000000	43	367	95522	0
2733	MT	Phillips County	0 986121	0 598523	0 002982	56	407	104314	1
2734	MT	Pondera County	0 990755	0 859667	0 021468	92	1007	95773	0
2735	MT	Powder River County	0 999770	0 772118	0 005222	39	272	103142	1
2736	MT	Powell County	0 997108	0 453822	0 003685	66	783	89752	1
2737	MT	Prairie County	0 996554	0 614604	0 003530	69	374	113173	1
2738	MT	Ravalli County	0 997462	0 157702	0 204360	234	3542	88898	0
2739	MT	Richland County	0 990999	0 897418	0 007293	114	560	101293	1
2740	MT	Roosevelt County	0 994073	0 938178	0 000000	65	507	82159	0
2741	MT	Rosebud County	0 997102	0 806075	0 003105	41	287	94642	1
2742	MT	Sanders County	0 990003	0 215573	0 044641	78	1004	75426	0
2743	MT	Sheridan County	0 982578	0 896923	0 004314	80	713	112279	1
2744	MT	Silver Bow County	0 999075	0 216968	0 000000	61	920	107003	0
2745	MT	Stillwater County	0 994473	0 774223	0 006449	86	624	112638	1
2746	MT	Sweet Grass County	0 996284	0 705712	0 003711	53	755	114765	1
2747	MT	Teton County	0 991291	0 810528	0 013902	122	786	106949	0
2748	MT	Toole County	0 982058	0 869257	0 008674	72	687	118545	1
2749	MT	Treasure County	0 994673	0 956133	0 000000	78	358	128786	0
2750	MT	Valley County	0 972118	0 535997	0 004940	59	550	105885	1
2751	MT	Wheatland County	0 996363	0 933867	0 032714	62	505	106312	0
2752	MT	Wibaux County	0 999124	0 862623	0.024030	51	370	97148	0
2753	MT	Yellowstone County	0 994754	0 862518	0 038621	170	751	116987	0
2754	MT	Yellowstone National Park	0 997044	0 000000	0 000000	0	0	89447	23
2755	ID	Ada County	0 994966	0 344889	0 185576	1031	3831	127241	0
2756	ID	Adams County	0 996044	0 253274	0 025961	120	1157	99431	0
2757	ID	Bannock County	0 970130	0 456640	0 106703	197	1256	90443	0
2758	ID	Bear Lake County	0 925622	0 433373	0 227533	131	1034	81224	0
2759	ID	Benewah County	0 989825	0 224519	0 011373	279	2266	94330	1
2760	ID	Bingham County	0 987974	1.023054	0 073475	388	1490	92007	9
2761	ID	Blaine County	0 993898	0 157315	0 045739	247	1906	157023	1
2762	ID	Boise County	0 997743	0 065976	0 000000	109	1710	77661	0
2763	ID	Bonner County	0 905189	0 134902	0 162822	99	2807	86981	0
2764	ID	Bonneville County	0 983084	0 379337	0 034346	554	2294	112184	0
2765	ID	Boundary County	0 992645	0 089484	0 055882	405	3305	77675	0
2766	ID	Butte County	0.999669	0 111512	0 046801	301	1388	106522	0
2767	ID	Camas County	0 996234	0 188212	0 000000	82	1040	130933	0
2768	ID	Canyon County	0 977201	0 1306039	0 107633	1657	4316	89068	9
2769	ID	Caribou County	0 981868	0 519938	0 091678	155	870	116912	0
2770	ID	Cassia County	0.994628	0 405658	0 054728	1054	2049	106840	0
2771	ID	Clark County	0 999671	0 253859	0 000000	316	878	178447	0
2772	ID	Clearwater County	0 989269	0 065536	0 077686	110	1427	97554	1
2773	ID	Custer County	0 997707	0 044633	0 005079	247	2268	115050	1
2774	ID	Elmore County	0 992609	0.179477	0.001090	1853	1321	91127	0
2775	ID	Franklin County	0 995627	0 540230	0 465390	483	1905	70550	0
2776	ID	Fremont County	0 984788	0 318829	0 027239	559	1878	95563	0
2777	ID	Gem County	0 994397	0 547613	0 161742	370	2415	86121	0
2778	ID	Gooding County	0 995852	0 485557	0 289474	2197	3178	96511	0
2779	ID	Idaho County	0 997931	0 137058	0 023206	99	1277	91506	0
2780	ID	Jefferson County	0 990523	0 444142	0 073231	724	2529	86426	0
2781	ID	Jerome County	0 996625	0 540606	0 367528	2075	4084	85356	0
2782	ID	Kootenai County	0 946365	0 164733	0 004637	321	4097	101083	0
2783	ID	Latah County	0 999779	0 503988	0 003607	282	2008	94276	1
2784	ID	Lemhi County	0 998830	0 066380	0 047223	238	1718	92833	0
2785	ID	Lewis County	0 998404	0 688319	0 003664	229	1775	134800	1
2786	ID	Lincoln County	0 999728	0 171634	0 246973	712	1955	117095	0
2787	ID	Madison County	0 996113	0 743454	0 040302	806	3654	74688	0
2788	ID	Minidoka County	0 995599	0 428145	0 075650	1534	4002	89393	0
2789	ID	Nez Perce County	0 991499	0 879294	0 002109	175	1456	110186	1
2790	ID	Oneida County	0 998938	0 352930	0 104641	120	990	87781	0
2791	ID	Owyhee County	0 997569	0 153033	0 090195	317	1129	84177	0
2792	ID	Payette County	0 993490	0 570426	0 214099	718	2533	83710	0
2793	ID	Power County	0.974331	0 483613	0.011509	559	1783	134231	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2794	ID	Shoshone County	0 999397	0 002627	0 199257	200	5121	93978	1
2795	ID	Teton County	0 999538	0 467606	0 101768	370	2823	91174	0
2796	ID	Twin Falls County	0 998218	0 397694	0 191315	860	2940	106204	0
2797	ID	Valley County	0 985095	0 033480	0 000000	204	2249	118145	0
2798	ID	Washington County	0 988310	0 596657	0 082910	153	1382	88560	0
2799	WY	Albany County	0 991834	0 683054	0 000000	44	387	92610	0
2800	WY	Big Horn County	0 993045	0 219808	0 007661	224	950	90009	0
2801	WY	Campbell County	0 999000	0 880832	0 000957	29	215	123293	1
2802	WY	Carbon County	0 991479	0 538386	0 000563	42	351	114393	0
2803	WY	Converse County	0 997560	0 867844	0 000765	26	275	93978	0
2804	WY	Crook County	0 995844	0 842970	0 002151	46	402	106312	1
2805	WY	Fremont County	0 990985	0 411078	0 016847	57	461	86223	0
2806	WY	Goshen County	0 996936	0 866769	0 001392	255	619	89176	0
2807	WY	Hot Springs County	0 998842	0 708210	0 000000	28	378	112930	0
2808	WY	Johnson County	0 997982	0 770863	0 000000	28	272	112997	0
2809	WY	Laramie County	0 999414	0 989040	0 007673	89	494	112130	0
2810	WY	Lincoln County	0 995130	0 214627	0 210819	111	1254	91309	0
2811	WY	Natrona County	0 993330	0 734060	0 004519	24	285	121626	0
2812	WY	Niobrara County	0 999206	0 800057	0 002808	40	271	110329	1
2813	WY	Park County	0 996253	0 179430	0 023543	177	912	110417	0
2814	WY	Platte County	0 987699	1 022868	0 017312	114	503	100792	9
2815	WY	Sheridan County	0 998526	0 748473	0 022268	66	662	122913	0
2816	WY	Sublette County	0 988988	0 189728	0 000000	108	996	114040	0
2817	WY	Sweetwater County	0 993722	0 257883	0 016092	11	328	118314	1
2818	WY	Teton County	0 949308	0 024290	0 013750	353	4358	198014	1
2819	WY	Uinta County	0 997169	0 660271	0 015324	47	473	82857	0
2820	WY	Washakie County	0 998801	0 277521	0 000000	205	726	104761	0
2821	WY	Weston County	0 999078	0 967365	0 000000	43	280	107944	0
2822	CO	Adams County	0 995151	0 898964	0 068682	308	1387	109157	0
2823	CO	Alamosa County	0 998916	0 448444	0.010039	540	2234	86311	0
2824	CO	Arapahoe County	0 997143	0 628029	0 004812	114	1273	152444	1
2825	CO	Archuleta County	0 996119	0 180017	0 000000	108	2238	80019	0
2826	CO	Baca County	0 999465	0 768591	0 001401	101	563	141479	1
2827	CO	Bent County	0 982338	0 822407	0 005530	161	381	104097	1
2828	CO	Boulder County	0 988100	0 331444	0 186684	1058	7993	145089	0
2829	CO	Chaffee County	0 998469	0 129764	0 018692	113	2627	97283	1
2830	CO	Cheyenne County	0 999987	0 801732	0 005716	102	638	201482	1
2831	CO	Clear Creek County	0 997442	0 028167	0 000000	9	3982	108798	18
2832	CO	Conejos County	0 997170	0 369696	0 030535	185	1368	66743	0
2833	CO	Costilla County	0 997286	0 421210	0 000000	102	965	86067	0
2834	CO	Crowley County	0 985836	0 839195	0 012072	552	475	82288	0
2835	CO	Custer County	0 998627	0 331559	0 016366	69	1480	102709	1
2836	CO	Delta County	0 994428	0 356667	0 089498	423	2543	90307	0
2837	CO	Denver County	0 989672	0 046589	0 000000	1023	5471	150067	12
2838	CO	Dolores County	0 998944	0 244704	0 000000	103	1078	107491	0
2839	CO	Douglas County	0 996869	0 430278	0 006652	116	4367	139792	1
2840	CO	Eagle County	0 997679	0 197172	0 000000	86	2461	139880	0
2841	CO	Elbert County	0 999940	0 933341	0 037969	75	868	124789	0
2842	CO	El Paso County	0 998583	0 629942	0 122556	76	862	113837	0
2843	CO	Fremont County	0 999332	0 338013	0 357260	100	1208	87557	0
2844	CO	Garfield County	0 997095	0 233557	0 009107	89	1978	109238	1
2845	CO	Gilpin County	0 997431	0 138612	0 000000	25	1680	97859	7
2846	CO	Grand County	0 989333	0 252687	0 007567	79	1412	117664	1
2847	CO	Gunnison County	0 993579	0 085545	0 016298	123	1950	92176	1
2848	CO	Hinsdale County	0 995142	0 012610	0 000000	163	3044	127837	0
2849	CO	Huerfano County	0 998512	0 630263	0 026779	31	720	85078	1
2850	CO	Jackson County	0 995225	0 457151	0 003862	98	1001	110525	1
2851	CO	Jefferson County	0 992324	0 209358	0 003494	492	6328	142217	1
2852	CO	Kiowa County	0 991732	0 774983	0 002800	72	536	215889	1
2853	CO	Kit Carson County	0 999680	0 970127	0 011206	319	960	148570	0
2854	CO	Lake County	0 981746	0 059740	0 000000	124	1563	69466	0
2855	CO	La Plata County	0 995446	0 542293	0 028074	60	1714	105269	0
2856	CO	Larimer County	0 987637	0 324587	0 220228	438	2900	116093	0
2857	CO	Las Animas County	0 999419	0 748666	0 019732	28	304	84110	0
2858	CO	Lincoln County	0 999884	1 002982	0 001977	80	499	137327	9
2859	CO	Logan County	0 996567	0 906293	0 004143	629	863	108811	0
2860	CO	Mesa County	0 996001	0 197305	0 074248	268	2554	102397	0
2861	CO	Mineral County	0 997737	0 027722	0 000000	35	2108	105276	10
2862	CO	Moffat County	0 998172	0 382119	0 004323	35	642	99322	1
2863	CO	Montezuma County	0 998458	0 639762	0 019483	44	1042	95102	1

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2864	CO	Montrose County	0 999125	0 311986	0 074717	304	2333	97243	0
2865	CO	Morgan County	0 993426	0 913482	0 032925	1139	1447	100230	0
2866	CO	Otero County	0 994575	0 783496	0 005272	400	866	94215	0
2867	CO	Ouray County	0 999644	0 343842	0 000000	62	2870	109807	0
2868	CO	Park County	0 995478	0 276106	0 000000	39	1086	104138	0
2869	CO	Phillips County	0 999830	1 044435	0 027636	444	1409	125067	9
2870	CO	Pitkin County	0 997114	0 051637	0 099327	167	4752	219181	1
2871	CO	Prowers County	0 997586	0 956611	0 000371	411	838	107145	0
2872	CO	Pueblo County	0 996227	0 586716	0.051247	99	764	96159	0
2873	CO	Rio Blanco County	0 999398	0 265111	0 000000	68	1002	115077	0
2874	CO	Rio Grande County	0 999415	0 376009	0.001656	489	2516	95827	1
2875	CO	Routt County	0 997285	0 381333	0 005458	113	1400	136758	1
2876	CO	Saguache County	0 999431	0 227860	0 004558	253	1232	75365	1
2877	CO	San Juan County	0 997780	0 001152	0 000000	112	6269	122019	2
2878	CO	San Miguel County	0 998460	0 243714	0 000000	54	1747	116343	0
2879	CO	Sedgwick County	0 997520	0 884568	0 000000	304	1251	134482	0
2880	CO	Summit County	0 982106	0 098822	0 000000	53	1485	152329	0
2881	CO	Teller County	0 996614	0 291720	0 000000	27	1316	102133	0
2882	CO	Washington County	0 998758	0 826492	0 005580	168	769	146477	0
2883	CO	Weld County	0 992792	0.816432	0 063029	1398	1905	107518	0
2884	CO	Yuma County	0 998703	0 946362	0 014395	692	1219	145448	0
2885	NM	Bernalillo County	0 997767	0 556396	0 555758	134	1228	118653	0
2886	NM	Catron County	0 999824	0 350315	0 038427	20	331	88837	1
2887	NM	Chaves County	0 999307	0 800956	0 513579	120	402	96044	0
2888	NM	Cibola County	0 999448	0 716189	0 000000	12	331	60214	0
2889	NM	Colfax County	0 996979	0 867314	0 013899	41	484	95732	1
2890	NM	Curry County	0 998779	1 027557	0 063399	375	947	95888	9
2891	NM	DeBaca County	0 996159	0 902690	0 019522	46	287	87232	1
2892	NM	Dona Ana County	0 998050	0 216027	0 316460	937	4625	77072	0
2893	NM	Eddy County	0 996295	0 425414	0 071182	89	504	96288	1
2894	NM	Grant County	0 999591	0 476426	0 000000	22	429	84218	0
2895	NM	Guadalupe County	0 999603	0 790315	0 000000	26	238	68789	0
2896	NM	Harding County	0 999713	0 948112	0 017313	27	296	88465	0
2897	NM	Hidalgo County	0 999919	0 382427	0 000000	55	414	88641	0
2898	NM	Lea County	0 999758	0 764469	0 208408	49	309	90951	0
2899	NM	Lincoln County	0 999960	0 608566	0 116820	16	295	95218	1
2900	NM	Los Alamos County	1 000000	0 000143	0 000000	5922	26626	188606	19
2901	NM	Luna County	0 999937	0 420222	0 000000	154	467	71220	0
2902	NM	McKinley County	0 998810	0 924485	0 142595	8	423	65483	1
2903	NM	Mora County	0 998769	0 732403	0 005838	24	456	59855	0
2904	NM	Otero County	0 999864	0 274922	0 148349	21	558	84712	1
2905	NM	Quay County	0 997615	0 961471	0 035871	57	383	88268	1
2906	NM	Rio Arriba County	0 993475	0 414190	0 003264	19	899	61372	0
2907	NM	Roosevelt County	0 997403	1 050750	0 261529	144	575	80533	9
2908	NM	Sandoval County	0 998656	0 324387	0 191367	49	831	87767	1
2909	NM	San Juan County	0 995609	0 537267	0 010437	67	826	80310	0
2910	NM	San Miguel County	0 996074	0 854458	0 001667	17	469	64054	0
2911	NM	Santa Fe County	0 999164	0 423860	0 121258	38	726	115774	1
2912	NM	Sierra County	0 986765	0 461139	0 111228	26	272	84482	20
2913	NM	Socorro County	0 999702	0 439120	0 202046	36	437	70441	0
2914	NM	Taos County	0 999338	0 230105	0 000000	20	1206	70719	10
2915	NM	Torrance County	0 999702	0 839597	0 000793	26	320	70333	0
2916	NM	Union County	0 999795	0 964551	0 000660	114	332	114657	0
2917	NM	Valencia County	0 999395	0 511100	0 513272	120	1415	76036	0
2918	AZ	Apache County	0 998798	0 806748	0 010278	6	299	53393	0
2919	AZ	Cochise County	0 992069	0 479043	0 033864	62	830	86812	1
2920	AZ	Coconino County	0 997673	0 502672	0 000000	6	450	86650	0
2921	AZ	Gila County	0 994155	0 377274	0 066308	10	707	80195	1
2922	AZ	Graham County	0 997451	0 623190	0 018394	47	636	67265	1
2923	AZ	Greenlee County	0 999235	0 116594	0 142216	81	837	88946	1
2924	AZ	La Paz County	0 996878	0 085438	0 000000	619	3841	81644	0
2925	AZ	Maricopa County	0 997744	0 123917	0 345155	1709	6802	122202	0
2926	AZ	Mohave County	0 988265	0 232623	0 053918	22	502	82606	1
2927	AZ	Navajo County	0 999356	1 134865	0 073380	7	413	65043	13
2928	AZ	Pima County	0 999735	0 590550	0 016601	27	401	102892	1
2929	AZ	Pinal County	0 999163	0 553554	0 047092	419	2461	77187	0
2930	AZ	Santa Cruz County	0 999617	0 421991	0 000000	35	1998	78230	0
2931	AZ	Yavapai County	0 999411	0 405617	0 000000	23	491	92664	0
2932	AZ	Yuma County	0 999115	0 064990	0 000000	4333	10323	77824	0
2933	UT	Beaver County	0 999107	0 115999	0 258715	261	971	84902	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
2934	UT	Box Elder County	0 850543	0 395826	0 210935	144	954	106481	0
2935	UT	Cache County	0.992718	0 359456	0 476985	811	3432	83243	0
2936	UT	Carbon County	0 995880	0 308427	0.026869	26	756	101612	1
2937	UT	Daggett County	0 965833	0 049126	0 000000	149	1557	113119	0
2938	UT	Davis County	0 480611	0 258399	0 060494	1673	10119	90720	0
2939	UT	Duchesne County	0 994504	0 192520	0 208575	138	1421	82938	0
2940	UT	Emery County	0 997828	0 084417	0 143224	84	1046	75420	0
2941	UT	Garfield County	0 993476	0 041529	0 032468	136	2144	86968	1
2942	UT	Grand County	0 996612	0 026786	0 069742	92	1452	90612	1
2943	UT	Iron County	0 998907	0 205672	0 071794	143	1110	74261	0
2944	UT	Juab County	0 995724	0 153254	0 026702	68	1027	72541	1
2945	UT	Kane County	0 971663	0 082121	0 001103	32	956	78176	0
2946	UT	Millard County	0 965016	0 114801	0 076239	304	1586	84604	0
2947	UT	Morgan County	0 997277	0.601721	0 301932	110	1199	99857	0
2948	UT	Piute County	0 989619	0 120656	0 431962	254	1701	75162	0
2949	UT	Rich County	0 946838	0 749001	0 000000	79	670	103562	0
2950	UT	Salt Lake County	0 912844	0 228124	0 185661	463	5559	104300	0
2951	UT	San Juan County	0.985767	0 064916	0 036415	68	790	55168	1
2952	UT	Sanpete County	0.990921	0 440233	0 160814	419	1431	72697	0
2953	UT	Sevier County	0 995815	0 129383	0 191349	479	1698	80601	0
2954	UT	Summit County	0 994156	0 311959	0 248744	99	1492	137394	0
2955	UT	Tooele County	0 953097	0 098358	0 034099	81	681	101374	1
2956	UT	Uintah County	0 995130	0 451828	0 127771	35	439	74864	0
2957	UT	Utah County	0 933375	0 352088	0 210068	481	2727	77668	0
2958	UT	Wasatch County	0 976589	0 184371	0 433816	137	3306	85363	0
2959	UT	Washington County	0 998819	0 107746	0 031495	100	2049	76679	0
2960	UT	Wayne County	0 997507	0 067045	0 228280	204	1409	68985	0
2961	UT	Weber County	0 872769	0 696362	0 412186	296	2350	103637	0
2962	NV	Churchill County	0 981228	0 084964	0 543783	285	2532	104219	0
2963	NV	Clark County	0 977711	0 016216	0 687062	567	5162	126151	0
2964	NV	Douglas County	0 962311	0 175286	0.267298	357	4228	159787	0
2965	NV	Elko County	0 998693	0 286443	0 023147	40	375	95638	21
2966	NV	Esmeralda County	0 999862	0 848768	0 000000	6	171	150751	0
2967	NV	Eureka County	0 998976	0 088238	0 000000	86	651	193903	0
2968	NV	Humboldt County	0 998969	0 119521	0 025716	153	652	116458	21
2969	NV	Lander County	0 995318	0 140586	0 000000	40	616	108825	0
2970	NV	Lincoln County	0 999741	0 007195	0 088103	337	2602	110274	21
2971	NV	Lyon County	0 988739	0 148292	0 141365	473	3295	112191	0
2972	NV	Mineral County	0 985170	0.112956	0 000000	26	1170	104890	0
2973	NV	Nye County	0 999314	0 012087	0 128053	243	1411	102743	21
2974	NV	Pershing County	0 990312	0 162411	0 000000	100	530	121633	0
2975	NV	Storey County	0 998697	0 261179	0 000000	4	1171	106983	0
2976	NV	Washoe County	0 968097	0 175064	0 039925	51	658	144452	21
2977	NV	White Pine County	0 997685	0 040795	0 000000	93	859	110640	0
2978	NV	Carson City	0 921043	0 058353	0 901479	350	4125	138444	22
2979	WA	Adams County	0 997550	0 809044	0 035262	548	1411	114447	0
2980	WA	Asotin County	0 992540	0 674588	0 041655	65	828	104456	1
2981	WA	Benton County	0 967576	0 587499	0 010974	825	2773	117393	0
2982	WA	Chelan County	0 975872	0 059943	0 000497	3351	9475	112361	1
2983	WA	Clallam County	0 652118	0 021715	0 361574	728	10352	111094	0
2984	WA	Clark County	0 956765	0 206463	0 373985	1233	11616	113722	0
2985	WA	Columbia County	0 994604	0 548398	0 000000	159	1749	121423	0
2986	WA	Cowlitz County	0 976259	0 048956	0 172934	1184	7374	109509	0
2987	WA	Douglas County	0 984766	0 787897	0 000692	294	1415	106630	1
2988	WA	Ferry County	0 976310	0 530352	0 000000	15	841	82735	0
2989	WA	Franklin County	0 981667	0 842956	0 037386	883	2714	99796	0
2990	WA	Garfield County	0 989359	0 715725	0.004078	141	1080	162442	1
2991	WA	Grant County	0 958803	0 634045	0 030083	1097	2821	105059	0
2992	WA	Grays Harbor County	0 861847	0 036463	0 639554	1026	7649	103400	0
2993	WA	Island County	0 403164	0 146264	0 723597	1171	12816	105018	0
2994	WA	Jefferson County	0 830484	0 008295	0 477198	863	8532	109590	0
2995	WA	King County	0 921679	0 031079	0 478072	4940	25276	161060	0
2996	WA	Kitsap County	0 699657	0 040649	0 007467	2538	24020	115205	0
2997	WA	Kittitas County	0 984606	0 241695	0 064546	489	2645	99519	0
2998	WA	Klickitat County	0 983294	0 575473	0 065735	122	1135	106380	0
2999	WA	Lewis County	0 988250	0 072851	0 431173	1358	6194	100961	0
3000	WA	Lincoln County	0 987820	0 990935	0 000925	138	1342	149823	1
3001	WA	Mason County	0 914392	0 017826	0 189327	270	7417	95787	1
3002	WA	Okanogan County	0 991142	0 382927	0 004366	265	1617	101307	1
3003	WA	Pacific County	0 796486	0 052326	0 309700	960	5466	102736	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
3004	WA	Pend Oreille County	0 982556	0 061765	0 170802	124	2967	88085	0
3005	WA	Pierce County	0 927427	0 054786	0 257821	3579	13954	109685	0
3006	WA	San Juan County	0 281669	0 183355	0 064236	142	9046	148658	1
3007	WA	Skagit County	0 903564	0 082906	0 332488	3716	10188	111439	0
3008	WA	Skamania County	0 983714	0 003814	0 000000	612	8919	92298	0
3009	WA	Snohomish County	0 951593	0 055432	0 566523	3267	15897	125609	0
3010	WA	Spokane County	0 990447	0 554349	0 113574	292	3009	110843	0
3011	WA	Stevens County	0 975466	0 344425	0 237458	106	1743	90686	0
3012	WA	Thurston County	0 939744	0 128707	0 333681	3202	11840	117258	0
3013	WA	Wahkiakum County	0 921708	0 074566	0 776003	981	6983	110119	0
3014	WA	Walla Walla County	0 977879	0 873859	0 013660	687	2077	111338	0
3015	WA	Whatcom County	0 846788	0 087067	0 685751	4429	12517	108385	0
3016	WA	Whitman County	0 991629	1 016123	0 001933	275	2284	105913	13
3017	WA	Yakima County	0 996368	0 596456	0 112248	1039	2508	104131	0
3018	OR	Baker County	0 993422	0 416934	0 083468	118	1015	99363	0
3019	OR	Benton County	0 996303	0 274431	0 123283	775	5409	111487	0
3020	OR	Clackamas County	0 994236	0 124485	0 043667	2510	13926	135071	0
3021	OR	Clatsop County	0 762640	0 046725	0 480708	655	4679	115022	0
3022	OR	Columbia County	0 954087	0 170910	0 185436	422	5899	104937	0
3023	OR	Coos County	0 886016	0 170716	0 292840	389	3374	100880	0
3024	OR	Crook County	0 997311	0 469273	0 009385	78	704	105079	1
3025	OR	Curry County	0 818365	0 071408	0 015396	347	3333	102492	0
3026	OR	Deschutes County	0 988006	0 072206	0 122066	290	4741	113905	0
3027	OR	Douglas County	0 981064	0 124713	0 006212	192	3285	99986	0
3028	OR	Gilliam County	0 984647	0 994457	0 003806	56	507	143802	1
3029	OR	Grant County	0 999843	0 398282	0 003629	39	575	108358	1
3030	OR	Harney County	0 990988	0 224678	0 001186	60	686	110112	0
3031	OR	Hood River County	0 978990	0 081360	0 001199	4989	17051	111223	1
3032	OR	Jackson County	0 994084	0 147114	0 022408	487	4667	105540	0
3033	OR	Jefferson County	0 994209	0 465846	0 001551	198	932	94506	1
3034	OR	Josephine County	0 998756	0 029779	0 474065	973	9910	95773	0
3035	OR	Klamath County	0 968779	0 189287	0 121985	292	1812	96843	0
3036	OR	Lake County	0 973359	0 159974	0 000137	108	862	111900	0
3037	OR	Lane County	0 964464	0 083070	0 101132	717	5920	106346	0
3038	OR	Lincoln County	0 820551	0 054694	0 056990	416	4550	107653	1
3039	OR	Linn County	0 992256	0 259435	0 099150	906	4744	95834	0
3040	OR	Malheur County	0 995694	0 208346	0 055264	374	1350	98374	0
3041	OR	Marion County	0 991459	0 398819	0 105344	2558	8698	108520	0
3042	OR	Morrow County	0 992274	0 860095	0 001400	208	980	127797	1
3043	OR	Multnomah County	0 934649	0 112338	0 001704	3053	13791	130147	1
3044	OR	Polk County	0 995809	0 353956	0 159264	966	5254	99817	0
3045	OR	Sherman County	0 990381	0 925268	0 003200	104	812	166993	1
3046	OR	Tillamook County	0 826954	0 056079	0 909728	3977	8748	101374	0
3047	OR	Umatilla County	0 995049	0 712687	0 002951	315	1377	102065	0
3048	OR	Union County	0 999157	0 363118	0 00065	242	1440	99119	0
3049	OR	Wallowa County	0 997983	0 344898	0 009187	102	857	118267	1
3050	OR	Wasco County	0 994040	0 756559	0 002703	104	1016	111765	1
3051	OR	Washington County	0 996377	0 301837	0 081889	2254	12354	135003	0
3052	OR	Wheeler County	0 999683	0 663384	0 013724	22	523	139061	0
3053	OR	Yamhill County	0 996093	0 392560	0 122395	1699	7736	103596	0
3054	CA	Alameda County	0.898010	0 606540	0 023127	369	6259	150013	1
3055	CA	Alpine County	0 993848	0 010086	0 000000	92	4007	155784	7
3056	CA	Amador County	0 980645	0 622879	0 000000	237	2999	101442	0
3057	CA	Butte County	0 977572	0.431076	0 010599	997	6301	101943	0
3058	CA	Calaveras County	0 983859	0 376887	0 022254	149	2624	92657	1
3059	CA	Colusa County	0 995204	0 611320	0 001915	943	4514	115219	1
3060	CA	Contra Costa County	0 897914	0 353670	0 030475	818	8050	172872	1
3061	CA	Del Norte County	0 819529	0 019524	0 230491	4118	9769	78908	0
3062	CA	El Dorado County	0 955446	0 093146	0 034544	231	9276	130384	1
3063	CA	Fresno County	0 990913	0 465003	0 070692	2898	7953	110843	0
3064	CA	Glenn County	0 990686	0 563167	0 156677	899	5127	106028	0
3065	CA	Humboldt County	0 881623	0 261422	0 497351	256	2369	109306	0
3066	CA	Imperial County	0 931498	0 199423	0 008049	3492	6090	98367	0
3067	CA	Inyo County	0 996512	0 037951	0 000000	60	3256	122249	0
3068	CA	Kern County	0 997497	0 544950	0 046348	1163	3171	107565	0
3069	CA	Kings County	0 998494	0 872432	0 277335	1853	4560	94195	0
3070	CA	Lake County	0 946507	0 203783	0 030778	483	5460	104930	1
3071	CA	Lassen County	0 965445	0 167135	0 020525	163	1319	86589	1
3072	CA	Los Angeles County	0 854314	0 070647	0 032256	2698	16573	140788	0
3073	CA	Madera County	0 993051	0 547617	0 080855	1556	5622	94127	0

County Code	State	County Name	FRCLND Absolute	FRMFRC Absolute	DPF Absolute	ASFP \$/ha	VFRM \$/ha	VNFRM \$/person	Notes
3074	CA	Marin County	0 627585	0 507667	0 665291	617	6688	254273	0
3075	CA	Mariposa County	0 992017	0 221945	0 000000	84	2094	109536	0
3076	CA	Mendocino County	0 904804	0 322857	0 044349	228	2938	109177	1
3077	CA	Merced County	0 978130	0 792902	0 311481	2291	6302	94561	0
3078	CA	Modoc County	0 938325	0 272095	0 005604	211	1635	96681	1
3079	CA	Mono County	0 972091	0 053013	0 000000	178	1734	127350	0
3080	CA	Monterey County	0 880893	0 645696	0 006188	2165	4936	131840	0
3081	CA	Napa County	0 956263	0 487676	0 002677	1291	13483	153535	1
3082	CA	Nevada County	0 982680	0 118243	0 361924	155	8734	121403	1
3083	CA	Orange County	0 833079	0 120183	0 000000	7581	31190	165266	0
3084	CA	Placer County	0 936127	0 153226	0 036849	641	10491	136602	1
3085	CA	Plumas County	0 977126	0 073118	0 000000	137	2154	110904	0
3086	CA	Riverside County	0 986903	0 091824	0 284468	4941	15561	116397	0
3087	CA	Sacramento County	0 969906	0 613297	0 163469	1366	7351	128244	0
3088	CA	San Benito County	0 998795	0 674963	0 018677	446	3147	109753	0
3089	CA	San Bernardino County	0 997781	0 100242	0 689341	1090	1894	111074	0
3090	CA	San Diego County	0 928977	0 192451	0 031484	2369	13637	132673	0
3091	CA	San Francisco County	0 201392	0 000234	0 000000	260519	2750264	193253	7
3092	CA	San Joaquin County	0 981118	0 875034	0 197842	2475	10704	104666	0
3093	CA	San Luis Obispo County	0 913922	0 626230	0 005475	345	3985	116641	0
3094	CA	San Mateo County	0 606032	0 199763	0 000000	4308	18726	195197	0
3095	CA	Santa Barbara County	0 722634	0 477564	0 021669	1154	6175	148347	0
3096	CA	Santa Clara County	0 989797	0 414643	0 015019	902	5861	170637	0
3097	CA	Santa Cruz County	0 733546	0 185432	0 002722	11296	24407	149180	1
3098	CA	Shasta County	0 983899	0 160179	0 048316	211	2689	110247	0
3099	CA	Sierra County	0 991067	0 090866	0 000000	85	2300	114799	0
3100	CA	Siskiyou County	0 990460	0 160902	0 045550	195	2638	104280	0
3101	CA	Solano County	0 913234	0 642096	0 044857	822	5754	116960	0
3102	CA	Sonoma County	0 891344	0 512617	0 235019	1342	12133	149383	0
3103	CA	Stanislaus County	0.986638	0 794180	0 287131	2918	9730	103210	0
3104	CA	Sutter County	0 989878	0 824827	0 003487	1651	9461	107890	0
3105	CA	Tehama County	0 996200	0 538395	0 100357	231	2301	82924	0
3106	CA	Trinity County	0 990983	0 057058	0 191245	37	895	93565	1
3107	CA	Tulare County	0 996883	0 438618	0 288025	2530	8046	99634	0
3108	CA	Tuolumne County	0 982909	0 096120	0 026905	220	2188	100406	1
3109	CA	Ventura County	0 835891	0 271368	0 001479	5147	17519	145082	1
3110	CA	Yolo County	0 989805	0 800826	0 002811	1115	5551	133615	1
3111	CA	Yuba County	0 979641	0 581876	0 029790	1142	6226	84089	0

#### Notes

- 1 See note explanation A.
- 2 See note explanations B, C, D, E
- 3 See note explanations B, C, D, E, K
- 4 See note explanations C, D, L
- 5 See note explanations B, C, D, E, L
- 6 See note explanations A, B
- 7 See note explanation C
- 8 See note explanations B, C.
- 9 See note explanation F
- 10 See note explanation E
- 11 See note explanations C, D, E.
- 12 See note explanation B.
- 13 See note explanations A, F.
- 14 See note explanations A, C, D
- 15 See note explanations B, E.
16. See note explanations B, G
- 17 See note explanations B, H
- 18 See note explanation D
19. See note explanations C, E, M.
- 20 See note explanations A, C, N
- 21 See note explanations A, J.
- 22 See note explanations A, I
- 23 See note explanation O

- 24. See note explanations A, O.
- 25. See note explanations B, C, D, E, O

Note explanations

- A. The amount of dairy sales for this county were not disclosed because of Bureau of Census confidentiality requirements. The dairy sales for this county were estimated by distributing the state's total amount of undisclosed dairy sales equally amongst the state's dairy farms that resided in counties with undisclosed dairy sales
- B. The amount of farm land for this county was not disclosed because of Bureau of Census confidentiality requirements. The farm land for this county was estimated by distributing the state's total amount of undisclosed farm land equally amongst the state's farms that resided in counties with undisclosed farm land.
- C. The value of farm land and buildings for this county were not disclosed because of Bureau of Census confidentiality requirements. The value of farm land and buildings for this county were estimated by distributing the state's total amount of undisclosed farm land and building value equally amongst the state's farms that resided in counties with undisclosed farm land and building values
- D. The value of farm machinery for this county was not disclosed because of Bureau of Census confidentiality requirements. The value of farm machinery for this county was estimated by distributing the state's total amount of undisclosed farm machinery value equally amongst the state's farms that resided in counties with undisclosed farm machinery values.
- E. The amount of farm sales for this county were not disclosed because of Bureau of Census confidentiality requirements. The farm sales for this county were estimated by distributing the state's total amount of undisclosed farm sales equally amongst the state's farms that resided in counties with undisclosed farm sales
- F. The value of FRMFCD was truncated to 1.000000. The resulting excess acreage was considered insignificant
- G. The value of FRMFCD was truncated to 1 000000. The resulting excess estimated undisclosed acreage, 121579 acres, was added to Kleberg County
- H. The estimated undisclosed acreage includes 121579 acres from El Paso County
- I. Estimated undisclosed dairy sales reduced to equal the same average value per farm as Douglas County. The excess undisclosed dairy sales were added to the other undisclosed county estimates.
- J. Estimated undisclosed dairy sales include the excess estimated undisclosed amount from Carson City
- K. Estimated undisclosed farm land and building and machinery values set so that farm land and building and machinery values per farm land area equal those of the average value of the surrounding counties, Franklin, Herkimer, and St. Lawrence. The excess undisclosed estimate farm land and building and machinery values were apportioned among Kings, Queens, and Richmond counties so that those county's undisclosed estimates would all have equal farm land and building and machinery values per farmland area
- L. Estimated undisclosed farm land and building and machinery values include excess undisclosed estimates from Hamilton County
- M. Estimated undisclosed farm land and building value per farm land were set equal to the average value of the surrounding counties, Sandoval, Santa Fe, and Rio Arriba. The excess estimated undisclosed farm land and building value was added to the Sierra County undisclosed estimate.
- N. Estimated undisclosed farm land and building value includes excess undisclosed estimate from Los Alamos County
- O. One or more counties reported per capita income figures cojointly with this county. The per capita income for all such cojoint counties was set to be equal. The list of counties reporting cojointly is as follows
  - Yellowstone National Park included with Park County,
  - Independent city of Charlottesville included with Albemarle County,
  - Independent cities of Clifton Forge and Covington included with Alleghany County;
  - Independent cities of Staunton and Waynesboro included with Augusta County,
  - Independent city of Bedford included with Bedford County;
  - Independent city of Lynchburg included with Campbell County,
  - Independent city of Galax included with Carroll County,

Independent cities of Colonial Heights and Petersburg included with Dinwiddie County,  
Independent cities of Fairfax and Falls Church included with Fairfax County,  
Independent city of Winchester included with Frederick County,  
Independent city of Emporia included with Greensville County,  
Independent city of South Boston included with Halifax County,  
Independent city of Martinsville included with Henry County,  
Independent city of Williamsburg included with James City County,  
Independent city of Radford included with Montgomery County,  
Independent city of Danville included with Pittsylvania County,  
Independent city of Hopewell included with Prince George County,  
Independent cities of Manassas and Manassas Park included with Prince William County,  
Independent city of Salem included with Roanoke County;  
Independent cities of Buena Vista and Lexington included with Rockbridge County,  
Independent city of Harrisonburg included with Rockingham County;  
Independent city of Franklin included with Southampton County;  
Independent city of Fredericksburg included with Spotsylvania County,  
Independent city of Bristol included with Washington County,  
Independent city of Norton included with Wise County;  
Independent city of Poquoson included with York County, and  
Menominee County included with Shawano County



## APPENDIX D — STATISTICAL ABSTRACTS

This appendix contains the following values and source information for the parameters used in estimating the value of non-farm assets (VNFRM, \$ / person)

### 1993 & 1994 Statistical Abstract of the U.S.

Reproducible Tangible Wealth (1990) :	2.5652E+13 \$
Urban and Built-Up Land (1987) :	77187303 acres
Urban and Built-Up Land (1992) :	91815303 acres
Urban and Built-Up Land (1990 Straight Line approx) :	85964103 acres
Median Housing Value (1990)	79100 \$
Total Farm Assets (1990)	1 0036E+12 \$
Farm Household Possessions (1990) :	4.6400E+10 \$
Population (1990) .	248709873 persons

### NUREG/CR-4551 Volume 2, Revision 1, Part 7

Typical Suburban Lot Size .	0.2 acres
Land - Fraction of Housing Value .	20 %
Average non-farm value for the U.S (1990) .	126632 \$/person

### 1994 County and City Data Book

U.S. Per Capita Income (1990) .	18696 \$/person
---------------------------------	-----------------

No. 754. Gross and Net Stock of Fixed Reproducible Tangible Wealth in  
Current and Constant (1987) Dollars

[In billions of dollars. As of December 31]

ITEM	1970	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	1990
<b>CURRENT DOLLARS</b>												
Gross stock, total	4,428	14,306	15,670	16,527	17,277	18,259	19,330	20,503	21,774	22,966	24,361	25,652
Private	2,689	9,364	10,387	11,005	11,477	12,082	12,747	13,536	14,388	15,078	16,022	16,894
Nonresidential equipment	679	2,389	2,723	2,939	3,067	3,214	3,374	3,599	3,775	4,010	4,257	4,543
Nonresidential structures	790	2,683	3,032	3,247	3,376	3,573	3,783	3,951	4,156	4,478	4,755	4,964
Residential	1,219	4,292	4,632	4,818	5,034	5,296	5,590	5,986	6,457	6,590	7,010	7,387
Government	1,060	2,979	3,141	3,237	3,368	3,568	3,772	3,916	4,083	4,306	4,516	4,713
Equipment	274	488	543	583	621	657	695	732	764	814	873	949
Structures	785	2,491	2,599	2,654	2,748	2,911	3,077	3,184	3,319	3,492	3,643	3,764
Federal	424	885	968	1,018	1,067	1,119	1,176	1,227	1,269	1,325	1,394	1,469
Military	276	493	535	566	601	638	678	715	738	776	818	873
State and local	636	2,094	2,173	2,219	2,301	2,449	2,596	2,690	2,814	2,981	3,122	3,244
Consumer durable goods	680	1,963	2,142	2,285	2,433	2,608	2,811	3,051	3,303	3,582	3,823	4,045
Net stock, total	2,708	8,619	9,374	9,792	10,172	10,734	11,367	12,063	12,803	13,458	14,245	14,936
Private	1,674	5,814	6,413	6,737	6,987	7,348	7,752	8,224	8,729	9,108	9,650	10,130
Government	662	1,790	1,875	1,921	1,991	2,105	2,224	2,311	2,414	2,542	2,665	2,775
Consumer durable goods	372	1,014	1,086	1,134	1,194	1,281	1,391	1,527	1,660	1,808	1,930	2,031
<b>CONSTANT (1987) DOLLARS</b>												
Gross stock, total	12,476	17,469	17,963	18,370	18,845	19,441	20,091	20,753	21,400	22,062	22,717	23,339
Private	8,131	11,585	11,936	12,217	12,523	12,909	13,322	13,728	14,112	14,504	14,886	15,245
Nonresidential equipment	1,839	3,009	3,126	3,206	3,285	3,396	3,518	3,633	3,737	3,855	3,979	4,093
Nonresidential structures	2,411	3,255	3,384	3,506	3,606	3,729	3,870	3,983	4,088	4,191	4,293	4,398
Residential	3,881	5,321	5,427	5,605	5,632	5,783	5,934	6,112	6,286	6,458	6,614	6,754
Government	3,001	3,544	3,603	3,648	3,705	3,770	3,849	3,934	4,026	4,115	4,204	4,307
Equipment	662	625	630	634	647	670	698	730	768	802	837	879
Structures	2,339	2,918	2,972	3,014	3,058	3,100	3,152	3,204	3,259	3,313	3,367	3,428
Federal	1,141	1,122	1,138	1,142	1,156	1,177	1,204	1,233	1,266	1,293	1,319	1,351
Military	723	641	643	644	652	668	691	716	743	767	786	813
State and local	1,860	2,421	2,465	2,506	2,549	2,593	2,645	2,701	2,760	2,822	2,885	2,956
Consumer durable goods	1,344	2,340	2,424	2,505	2,617	2,762	2,920	3,091	3,262	3,444	3,627	3,787
Net stock, total	7,691	10,524	10,755	10,902	11,112	11,441	11,823	12,214	12,577	12,942	13,293	13,598
Private	5,074	7,198	7,384	7,499	7,642	7,866	8,112	8,346	8,558	8,774	8,980	9,157
Government	1,884	2,123	2,148	2,166	2,191	2,223	2,269	2,320	2,380	2,430	2,482	2,537
Consumer durable goods	733	1,203	1,223	1,236	1,278	1,352	1,442	1,547	1,639	1,738	1,831	1,903

Source: U.S. Bureau of Economic Analysis, Survey of Current Business,  
January and August 1992.

## No 359 Land Cover/Use, by State: 1987

[In thousands of acres Excludes Alaska and District of Columbia]

REGION, DIVISION, AND STATE	Total surface area <sup>1</sup>	Federal land	Rural land			Rural land				
			Total	Developed <sup>2</sup>	Total	Crop land	Pasture land	Range- land	Forest land	Minor cover/uses
United States	1,937,726	404,069	1,484,156	77,305	1,406,851	422,416	129,021	401,685	393,904	59,826
Northeast	108,080	2,392	100,778	9,611	91,166	14,532	7,669	0	64,246	4,720
New England	42,670	1,332	38,609	3,005	35,604	2,311	1,248	0	30,307	1,739
Maine	21,290	161	19,517	508	19,009	943	419	0	16,933	714
New Hampshire	5,938	729	4,971	372	4,599	163	115	0	4,052	269
Vermont	6,153	335	5,556	208	5,348	653	388	0	4,184	122
Massachusetts	5,302	89	4,849	1,063	3,786	291	179	0	2,937	379
Rhode Island	776	4	661	161	500	22	37	0	404	37
Connecticut	3,212	14	3,056	693	2,362	239	110	0	1,797	218
Middle Atlantic	65,410	1,060	62,168	6,606	55,562	12,221	6,422	0	33,939	2,981
New York	31,429	234	29,782	2,485	27,297	5,774	3,686	0	16,650	1,187
New Jersey	4,984	148	4,563	1,325	3,239	673	229	0	1,890	447
Pennsylvania	28,997	677	27,823	2,796	25,027	5,774	2,507	0	15,398	1,348
Midwest	490,474	17,883	460,128	23,789	436,339	233,455	40,718	71,859	71,637	18,670
East North Central	159,066	6,269	148,602	12,368	136,233	72,743	12,982	0	42,482	8,027
Ohio	26,451	347	25,686	2,925	22,762	12,537	2,444	0	6,426	1,354
Indiana	23,159	487	22,302	1,780	20,522	13,930	2,073	0	3,698	821
Illinois	36,061	492	34,792	2,792	32,000	25,121	2,689	0	3,447	744
Michigan	37,457	3,130	33,051	2,921	30,130	9,484	2,735	0	15,483	2,429
Wisconsin	35,938	1,813	32,770	1,951	30,820	11,671	3,041	0	13,428	2,680
West North Central	331,408	11,615	311,527	11,421	300,105	160,713	27,736	71,859	29,155	10,643
Minnesota	54,017	3,390	47,077	2,136	44,941	22,990	3,425	157	13,952	4,417
Iowa	36,016	172	35,387	1,688	33,699	27,031	3,866	0	1,841	961
Missouri	44,606	2,060	41,655	2,165	39,491	15,090	12,606	56	10,959	781
North Dakota	45,250	1,882	42,255	1,242	41,013	28,064	1,206	9,933	428	1,382
South Dakota	49,354	2,873	45,467	1,064	44,403	17,819	2,354	22,152	565	1,513
Nebraska	49,507	652	48,218	1,250	46,967	20,601	1,957	22,900	728	782
Kansas	52,658	587	51,467	1,876	49,592	29,119	2,324	16,660	681	808
South	575,044	26,391	527,041	30,657	496,384	107,532	67,937	113,837	189,507	17,571
South Atlantic	178,469	12,757	156,196	13,346	142,849	26,495	16,165	3,592	88,408	8,190
Delaware	1,309	33	1,213	165	1,048	521	30	0	357	141
Maryland	6,695	159	6,048	936	5,111	1,795	514	0	2,415	388
Virginia	26,091	2,368	22,812	1,663	21,150	3,309	3,315	0	13,622	904
West Virginia	15,508	1,116	14,227	532	13,695	1,053	1,892	0	10,466	284
North Carolina	33,708	2,309	28,622	2,487	26,135	6,548	1,992	0	16,528	1,067
South Carolina	19,912	1,340	17,785	1,422	16,363	3,371	1,177	0	11,073	742
Georgia	37,702	2,062	34,664	2,375	32,289	6,307	3,040	0	21,860	1,083
Florida	37,545	3,369	30,825	3,766	27,059	3,592	4,205	3,592	12,088	3,583
East South Central	116,446	5,116	108,068	5,705	102,363	22,870	18,493	96	58,115	2,789
Kentucky	25,862	1,169	24,023	1,224	22,799	5,818	5,955	0	10,054	972
Tennessee	26,972	1,369	24,759	1,669	23,090	5,765	5,019	0	11,601	706
Alabama	33,091	904	31,230	1,640	29,591	4,210	3,595	96	21,017	673
Mississippi	30,521	1,674	28,056	1,172	26,884	7,078	3,924	0	15,443	439
West South Central	280,129	8,518	262,778	11,606	251,172	58,167	33,279	110,149	42,984	6,593
Arkansas	34,040	3,129	29,904	1,232	28,672	8,182	5,678	164	14,268	380
Louisiana	30,561	1,174	26,472	1,455	25,016	6,484	2,276	234	12,736	3,286
Oklahoma	44,772	1,176	42,431	1,716	40,715	11,557	7,590	14,546	6,505	517
Texas	170,756	3,040	163,971	7,203	156,768	31,944	17,735	95,204	9,476	2,410
West	764,128	357,403	396,209	13,247	382,962	66,896	12,697	215,989	68,514	18,865
Mountain	552,680	266,171	280,033	5,964	274,069	44,235	7,828	182,653	27,532	11,822
Montana	94,109	27,074	65,682	999	64,682	17,881	3,169	36,769	5,253	1,611
Idaho	53,481	33,190	19,628	477	19,152	6,532	1,354	6,596	4,071	600
Wyoming	62,598	29,457	32,576	501	32,075	2,362	928	26,784	984	1,017
Colorado	66,618	23,833	42,320	1,375	40,945	10,967	1,266	23,427	4,079	1,207
New Mexico	77,819	26,423	51,144	698	50,445	2,297	186	40,782	4,685	2,496
Arizona	72,960	30,647	41,994	1,116	40,878	1,306	81	31,867	4,912	2,712
Utah	54,336	35,476	16,440	465	15,975	2,002	563	8,507	3,194	1,711
Nevada	70,759	60,071	10,250	333	9,916	889	282	7,921	356	469
Pacific	211,448	91,232	116,176	7,283	108,893	22,662	4,869	33,337	40,982	7,043
Washington	43,608	12,471	29,947	1,564	28,383	7,758	1,421	5,574	12,634	997
Oregon	62,127	32,305	28,918	941	27,977	4,348	1,916	9,152	11,857	705
California	101,572	46,014	53,654	4,621	49,033	10,209	1,501	17,719	15,073	4,531
Hawaii	4,141	443	3,657	157	3,500	348	31	891	1,419	811

<sup>1</sup> Includes water area not shown separately<sup>2</sup> Includes urban and built-up areas in units of 10 acres or greater, and rural transportation

Source: U S Dept of Agriculture, Soil Conservation Service, and Iowa State University, Statistical Laboratory, Statistical Bulletin No 790, Summary Report, 1987 National Resources Inventory, December 1989

No. 1239. Occupied Housing Units - Housing Value and Contract Rent, by State: 1990  
 (In thousands of units, except as indicated. As of April. Based on the Census of Population and Housing; see Appendix III)

STATE	SPECIFIED OWNER OCCUPIED UNITS												SPECIFIED RENTER OCCUPIED UNITS											
	VALUE CATEGORIES												Rent Categories											
	Total housing units	Less than \$50,000	\$50,000 to \$99,999	\$100,000 to \$149,999	\$150,000 to \$199,999	\$200,000 to \$299,999	\$300,000 or more	Lower quartile value <sup>1</sup> (dol.)	Median (dol.)	Upper quartile value <sup>1</sup> (dol.)	Total housing units	Less than \$200	\$200 to \$299	\$300 to \$399	\$400 to \$499	\$500 to \$599	\$600 to \$749	\$750 or more	No cash rent	Median (dol.)				
United States	44,918.0	11,402.5	16,957.5	6,773.3	4,017.2	3,376.9	2,390.7	49,500	79,100	137,800	32,170.0	2,815.1	3,736.2	5,879.2	5,935.0	4,407.0	4,064.4	3,913.8	1,419.3	447				
Alabama	753.8	343.9	310.7	62.5	20.1	11.3	5.4	35,500	53,700	78,800	428.0	81.6	87.3	102.1	67.2	29.1	15.7	6.8	38.1	325				
Alaska	77.5	11.0	31.9	23.0	7.5	3.2	0.9	66,400	94,400	128,500	81.9	1.4	4.4	9.8	12.9	11.3	13.1	17.9	11.1	559				
Arizona	668.7	98.7	372.7	118.8	40.6	23.9	14.0	59,800	80,100	109,900	485.8	25.5	46.1	111.5	113.1	71.2	53.0	39.8	25.6	438				
Arkansas	427.7	235.6	158.9	23.3	6.8	3.5	1.6	31,100	46,300	68,700	258.1	38.6	57.8	64.8	39.3	17.2	8.5	4.3	27.6	328				
California	4,690.3	119.0	636.6	812.1	851.5	1,151.0	1,120.0	127,100	195,500	294,800	4,553.4	136.2	198.5	379.5	621.9	736.4	1,004.1	1,361.9	114.9	620				
Colorado	637.6	84.1	363.1	122.8	37.4	19.9	10.3	62,500	82,700	109,700	472.6	36.5	63.3	110.0	96.4	62.2	51.9	36.0	16.3	418				
Connecticut	643.5	4.6	36.7	165.3	188.0	98.5	139,000	177,800	246,000	418.5	31.2	21.4	31.1	51.2	68.2	94.1	106.3	14.9	598					
Delaware	137.5	11.8	56.8	38.5	15.8	10.5	4.3	74,800	100,100	143,900	72.2	6.0	4.4	8.7	16.0	16.0	10.9	6.7	3.4	495				
District of Columbia	71.5	1.7	25.2	15.0	6.9	8.4	14.4	86,700	123,900	258,700	152.1	14.7	11.0	25.5	29.5	21.7	20.3	26.6	2.7	479				
Florida	2,378.2	433.1	1,239.1	381.9	151.5	100.6	72.0	56,100	77,100	112,200	1,669.6	103.2	130.9	271.1	368.9	308.6	253.3	168.5	65.1	481				
Georgia	1,138.8	314.5	531.2	163.2	68.9	39.9	23.1	47,300	71,300	102,100	808.4	97.8	99.6	134.3	155.9	128.1	98.0	53.2	41.4	433				
Hawaii	144.4	3.3	13.1	16.7	21.1	39.7	50.5	156,800	245,300	358,800	162.8	9.1	7.5	12.6	15.6	17.5	26.2	52.3	22.1	650				
Idaho	177.3	66.8	89.4	14.2	3.7	2.0	1.2	42,100	58,200	79,700	102.4	14.0	24.5	26.3	16.3	6.5	4.0	2.2	8.6	330				
Illinois	2,084.7	525.1	788.6	403.2	183.7	114.5	71.6	49,800	80,900	128,300	1,470.4	128.7	159.0	281.6	306.8	229.8	185.6	134.3	44.5	445				
Indiana	1,137.8	514.3	484.0	92.8	27.0	13.7	6.0	36,500	53,900	78,200	589.9	61.5	104.8	156.4	128.6	65.9	29.3	14.1	29.3	374				
Iowa	566.6	317.8	209.7	27.7	7.0	3.3	1.1	30,200	45,900	65,900	285.7	40.5	66.1	78.7	50.7	22.1	10.8	4.6	14.3	336				
Kansas	500.6	238.0	199.6	41.1	12.3	6.4	3.1	32,400	52,200	78,000	289.8	32.3	54.8	69.0	57.7	30.4	19.3	10.3	15.9	372				
Kentucky	662.2	326.9	260.5	47.5	15.9	8.0	3.4	33,400	50,500	73,400	392.3	69.2	87.8	100.6	56.9	24.1	11.9	5.4	36.4	319				
Louisiana	733.9	289.1	344.5	63.0	19.8	11.5	6.1	38,200	58,500	83,000	501.3	70.7	93.1	123.5	88.5	43.6	24.3	14.4	43.1	352				
Maine	214.7	37.5	95.2	49.3	18.0	10.0	4.7	60,100	87,400	123,300	133.3	15.5	14.2	26.6	26.6	19.2	14.1	7.5	9.7	419				
Maryland	970.9	87.3	300.9	274.2	139.6	103.3	65.6	79,000	116,500	169,300	598.3	41.2	36.0	65.4	94.0	106.0	128.8	108.2	18.8	548				
Massachusetts	1,004.6	9.5	93.5	310.8	298.3	196.4	96.1	126,800	162,800	216,000	910.0	98.4	58.2	74.9	106.8	130.2	184.5	231.1	26.0	580				
Michigan	1,916.1	737.2	814.5	219.2	79.3	46.0	20.0	38,500	60,600	90,900	966.2	82.1	123.4	209.0	223.3	149.7	92.0	54.6	32.1	423				
Minnesota	894.3	204.9	482.3	138.2	38.6	21.1	9.2	52,200	74,000	98,200	445.9	54.6	54.7	65.1	94.3	66.0	50.2	28.4	12.6	422				
Mississippi	441.8	249.3	156.1	24.2	7.0	3.6	1.6	31,400	45,600	67,800	247.5	49.8	54.0	54.0	32.7	15.7	7.9	3.4	29.9	309				
Missouri	1,005.4	385.1	452.6	102.0	33.6	20.3	11.9	38,800	59,800	87,100	585.0	74.0	108.7	138.7	115.2	62.5	37.6	18.1	30.1	368				
Montana	132.4	52.3	68.3	8.4	2.1	1.0	0.3	40,000	56,600	74,200	93.9	16.1	24.4	24.8	12.9	5.5	2.1	0.9	7.2	311				
Nebraska	314.4	155.3	132.4	18.5	4.7	2.4	1.0	32,800	50,400	70,400	186.6	24.6	39.8	48.6	34.8	16.2	7.7	4.6	10.3	348				
Nevada	183.8	8.4	93.3	51.3	16.0	9.1	5.6	75,400	95,700	129,400	209.2	8.4	11.5	29.0	48.5	44.4	38.9	22.9	5.5	509				
New Hampshire	199.4	6.3	45.6	79.9	39.8	21.1	6.6	98,400	129,400	167,600	127.8	7.7	7.1	12.3	21.6	24.9	27.3	21.5	5.4	549				
New Jersey	1,466.3	45.5	241.4	348.8	367.3	301.5	161.9	112,300	162,300	225,000	973.7	60.8	45.3	81.1	133.1	166.0	228.3	232.7	26.4	592				
New Mexico	262.3	73.5	128.2	37.3	12.8	7.2	3.3	46,700	70,100	97,800	173.1	18.3	28.6	44.9	31.0	16.4	11.9	7.4	14.4	372				
New York	2,387.6	243.8	674.4	431.6	439.0	381.0	217.8	76,200	131,600	200,400	3,150.6	242.8	281.8	490.1	605.9	463.7	454.7	532.9	78.5	486				
North Carolina	1,218.0	382.8	575.7	155.2	58.3	33.1	15.0	44,700	65,800	94,900	777.9	83.5	121.5	187.4	162.2	90.6	49.8	22.3	60.6	382				
North Dakota	103.7	50.6	46.3	5.1	1.1	0.5	0.1	30,800	50,800	70,000	78.5	13.8	18.7	19.8	11.4	4.0	1.8	0.7	8.3	313				
Ohio	2,241.3	734.0	1,102.8	260.5	80.8	42.9	20.3	43,900	63,500	90,200	1,293.4	149.2	206.4	341.2	280.5	148.7	79.0	40.8	49.7	379				
Oklahoma	616.3	323.7	235.8	36.8	10.9	6.1	3.3	30,800	48,100	71,000	370.7	44.2	85.0	98.7	60.5	30.1	17.8	9.0	25.4	340				
Oregon	511.8	133.3	281.2	62.7	19.5	10.5	4.7	49,200	67,100	92,600	394.9	30.2	52.5	99.2	96.1	52.1	33.7	18.1	13.1	408				
Pennsylvania	2,581.3	829.2	1,017.8	395.9	180.6	109.1	48.7	42,500	69,700	109,500	1,287.7	141.0	189.7	272.2	245.0	169.6	125.3	85.4	59.6	404				
Rhode Island	178.5	3.0	31.0	78.4	35.6	19.4	9.0	108,000	133,500	171,800	152.0	17.9	12.1	18.0	29.1	29.2	25.3	15.7	4.8	489				
South Carolina	615.4	225.7	280.1	63.5	23.7	14.8	7.6	41,100	61,100	89,100	368.9	45.3	59.5	83.9	77.1	38.5	21.2	10.4	33.0	376				
South Dakota	113.1	64.2	42.4	4.6	1.0	0.5	0.2	27,300	45,200	64,500	81.2	16.2	19.4	18.7	11.6	4.5	2.3	0.9	7.5	306				
Tennessee	938.4	371.8	418.4	91.5	32.2	17.8	8.7	39,600	58,400	85,300	568.9	86.8	101.6	134.5	110.3	55.1	27.6	13.9	39.3	357				
Texas	2,949.1	1,151.2	1,261.1	314.8	111.3	66.5	44.2	38,300	59,600	89,200	2,332.9	179.0	345.2	614.2	480.6	263.7	193.8	139.7	116.6	395				
Utah	303.7	61.1	188.6	35.2	10.6	5.7	2.5	53,100	69,900	92,000	168.8	12.4	31.3	53.8	33.9	15.7	9.6	6.3	6.8	369				
Vermont	89.2	7.9	41.3	25.5</td																				

No. 1104. Balance Sheet of the Farming Sector  
in Current and Constant (1987) Dollars

[In billions of dollars, except as indicated. As of December 31.  
Includes farm operator households.  
See Historical Statistics, Colonial Times to 1970,  
series K 204-219, for data before revisions]

ITEM	1970	1975	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989
CURRENT DOLLARS												
Assets	324.3	579.5	1,089.0	1,088.9	1,056.2	1,063.6	975.7	892.8	847.7	911.3	951.5	985.8
Physical assets:												
Real estate	224.5	421.0	850.1	851.7	819.1	829.3	735.0	657.0	613.0	658.6	682.2	703.9
Nonreal estate	76.6	127.2	199.6	196.3	194.4	189.6	193.0	186.5	181.0	196.1	212.1	223.6
Livestock and poultry <sup>1</sup>	23.7	29.4	60.6	53.5	53.0	49.5	49.5	46.3	47.8	58.0	62.2	66.2
Machinery, motor vehicles	34.4	63.1	86.9	92.5	92.6	92.1	91.1	88.3	86.1	84.5	86.1	89.2
Crops stored	8.5	20.5	32.7	29.5	25.8	23.6	26.1	22.9	16.3	17.5	23.3	23.4
Household furnishings, equipment	10.0	14.2	19.4	20.8	23.0	24.4	24.3	27.8	28.7	32.9	37.0	42.2
Purchased inputs	(NA)	(NA)	(NA)	(NA)	(NA)	(NA)	2.0	1.2	2.1	3.2	3.5	2.6
Financial assets	23.2	31.4	39.3	40.9	42.8	44.6	47.7	49.3	53.8	56.7	57.2	58.3
Investment in cooperatives	7.2	13.0	19.3	20.6	21.9	22.8	24.3	24.3	24.4	25.3	25.1	26.3
Other financial assets	16.0	18.4	20.0	20.3	20.9	21.8	23.4	25.0	29.4	31.4	32.1	32.0
Claims	324.3	579.5	1,089.0	1,088.9	1,056.2	1,063.6	975.7	892.8	847.7	911.3	951.5	985.8
Debt	52.8	91.5	178.7	195.4	203.1	206.5	204.3	187.9	166.6	153.7	148.5	146.0
Real estate debt	30.5	49.9	97.5	107.2	111.3	113.7	112.3	105.7	95.9	87.7	83.0	80.5
Nonreal estate debt	22.3	41.6	81.2	88.2	91.8	92.7	92.0	82.2	70.8	66.0	65.6	65.5
Equity	271.5	488.0	910.3	893.5	853.1	857.1	771.4	704.9	681.0	757.6	802.9	839.8
Farm debt/asset ratio (percent)	16.3	15.8	16.4	17.9	19.2	19.4	20.9	21.0	19.7	16.9	15.6	14.8
CONSTANT (1987) DOLLARS <sup>12</sup>												
Assets	923.9	1,177.7	1,518.8	1,380.1	1,260.4	1,219.7	1,072.2	945.8	874.8	911.3	915.8	908.6
Debt	150.3	185.9	249.3	247.7	242.4	236.8	224.5	199.0	171.9	153.7	142.9	134.6
Equity	773.6	991.8	1,269.6	1,132.4	1,018.0	982.9	847.7	746.7	702.8	757.6	772.8	774.0

NA Not available.

<sup>1</sup> Excludes horses and mules.

<sup>12</sup> Constant dollar figures are based on gross domestic product implicit price deflators for year.

Source: U.S. Dept. of Agriculture, Economic Research Service,  
Economic Indicators of the Farm Sector:  
National Financial Summary, 1991.



## APPENDIX E — OTHER CENSUS INFORMATION

This appendix contains additional census information that can be found on the PL 94-171 CD-ROMs. Included in this appendix is *APPENDIX A — Area Classifications*, *APPENDIX C — Accuracy of the Data*, *APPENDIX D — Collection and Processing Procedures*, and the PL 94-171 database *Data Dictionary*.

Census of Population and Housing, 1990: Public Law (P.L.) 94-171 Data on CD-ROM (Name of State) [machine-readable data files] / prepared by the Bureau of the Census. --Washington: The Bureau [producer and distributor], 1991.

### APPENDIX A.

#### Area Classifications

#### CONTENTS

Alaska Native Regional Corporation (See American Indian and Alaska Native Area)	
Alaska Native Village Statistical Area (See American Indian and Alaska Native Area)	
American Indian and Alaska Native Area.....	A-1
American Indian Reservation (See American Indian and Alaska Native Area, see County Subdivision)	
American Indian Reservation and Trust Land (See American Indian and Alaska Native Area)	
Area Measurement.....	A-3
Assessment District (See County Subdivision)	
Block.....	A-3
Block Group (BG).....	A-3
Block Numbering Area (BNA) (See Census Tract and Block Numbering Area)	
Borough (See County Subdivision, see Place)	
Borough and Census Area (Alaska) (See County)	
Boundary Changes.....	A-4
Census Area (Alaska) (See County)	
Census Block (See Block)	
Census Code (See Geographic Code)	
Census County Division (See County Subdivision)	
Census Designated Place (CDP) (See Place)	
Census Geographic Code (See Geographic Code)	
Census Subarea (Alaska) (See County Subdivision)	
Census Tract and Block Numbering Area.....	A-4
City (See Place)	
Congressional District.....	A-5
Consolidated City (See Place)	
County.....	A-5
County Subdivision.....	A-5
Crews of Vessels (See Area Measurement, see Block, see Census Tract and Block Numbering Area)	
Division (See County Subdivision)	
Election District (See County Subdivision, see Voting District)	
Federal Information Processing Standards (FIPS) Code (See Geographic Code)	
Geographic Block Group (See Block Group)	
Geographic Code.....	A-6
Geographic Presentation.....	A-6
Gore (See County Subdivision)	

Grant (See County Subdivision)	
Hierarchical Presentation (See Geographic Presentation)	
Historic Areas of Oklahoma (See American Indian and Alaska Native Area, Tribal Jurisdiction Statistical Area)	
Incorporated Place (See Place)	
Independent City (See County)	
Internal Point.....	A-7
Inventory Presentation (See Geographic Presentation)	
Land Area (See Area Measurement)	
Latitude (See Internal Point)	
Longitude (See Internal Point)	
Magisterial District (See County Subdivision)	
Minor Civil Division (See County Subdivision)	
Parish (Louisiana) (See County)	
Parish Governing Authority District (See County Subdivision)	
Place.....	A-7
Plantation (See County Subdivision)	
Precinct (See County Subdivision, see Voting District)	
Purchase (See County Subdivision)	
State.....	A-8
Supervisors' District (See County Subdivision)	
Tabulation Block Group (See Block Group)	
TIGER.....	A-8
Town (See County Subdivision, see Place)	
Township (See County Subdivision)	
Tract (See Census Tract and Block Numbering Area)	
Tribal Designated Statistical Area (TDSA) (See American Indian and Alaska Native Area)	
Tribal Jurisdiction Statistical Area (TJSA) (See American Indian and Alaska Native Area)	
Trust Land (See American Indian and Alaska Native Area)	
United States.....	A-8
United States Postal Service (USPS) Code (See Geographic Code)	
Unorganized Territory (unorg.) (See County Subdivision)	
Village (See Place)	
Voting District (VTD).....	A-9
Water Area (See Area Measurement)	

#### AMERICAN INDIAN AND ALASKA NATIVE AREA

##### Alaska Native Regional Corporation

Alaska Native Regional Corporations (ANRC's) are corporate entities established under the Alaska Native Claims Settlement Act of 1972, Public Law 92-203, as amended by Public Law 94-204, to conduct both business and nonprofit affairs of Alaska Natives. Alaska is divided into 12 ANRC's that cover the entire State, except for the Annette Islands Reserve. The boundaries of the 12 ANRC's were established by the Department of the Interior, in cooperation with Alaska Natives. Each ANRC was designed to include, as far as practicable, Alaska Natives with a common heritage and common interests. The ANRC boundaries for the 1990 census were identified by the Bureau of Land Management. A 13th region was established for Alaska Natives who are not permanent residents and who chose not to enroll in one of the 12 ANRC's; no census products are prepared for the 13th region. ANRC's were first identified for the 1980 census.

Each ANRC is assigned a two-digit census code ranging from 07 through 84. These census codes are assigned in alphabetical order of the

ANRC's.

#### Alaska Native Village Statistical Area

Alaska Native villages (ANV's) constitute tribes, bands, clans, groups, villages, communities, or associations in Alaska that are recognized pursuant to the Alaska Native Claims Settlement Act of 1972, Public Law 92-203. Because ANV's do not have legally designated boundaries, the Census Bureau has established Alaska Native village statistical areas (ANVSA's) for statistical purposes. For the 1990 census, the Census Bureau cooperated with officials of the nonprofit corporation within each participating Alaska Native Regional Corporation (ANRC), as well as other knowledgeable officials, to delineate boundaries that encompass the settled area associated with each ANV. ANVSA's are located within ANRC's and do not cross ANRC boundaries. ANVSA's for the 1990 census replace the ANV's that the Census Bureau recognized for the 1980 census.

Each ANVSA is assigned a four-digit census code ranging from 6001 through 8989. Each ANVSA also is assigned a five-digit FIPS code. Both the census and FIPS codes are assigned in alphabetical order of ANVSA's.

#### American Indian Reservation and Trust Land

American Indian Reservation--Federal American Indian reservations are areas with boundaries established by treaty, statute, and/or executive or court order, and recognized by the Federal Government as territory in which American Indian tribes have jurisdiction. State reservations are lands held in trust by State governments for the use and benefit of a given tribe. The reservations and their boundaries were identified for the 1990 census by the Bureau of Indian Affairs (BIA), Department of Interior (for Federal reservations), and State governments (for State reservations). The names of American Indian reservations recognized by State governments, but not by the Federal Government, are followed by "(State)." Areas composed of reservation lands that are administered jointly and/or are claimed by two reservations, as identified by the BIA, are called "joint areas," and are treated as separate American Indian reservations for census purposes.

Federal reservations may cross State boundaries, and Federal and State reservations may cross county, county subdivision, and place boundaries. For reservations that cross State boundaries, only the portion of the reservations in a given State are shown in the data products for that State; the entire reservations are shown in data products for the United States.

Each American Indian reservation is assigned a four-digit census code ranging from 0001 through 4989. These census codes are assigned in alphabetical order of American Indian reservations nationwide, except that joint areas appear at the end of the code range. Each American Indian reservation also is assigned a five-digit FIPS code; because the FIPS codes are assigned in alphabetical sequence of American Indian reservations within each State, the FIPS code is different in each State for reservations in more than one State.

Trust Land--Trust lands are property associated with a particular American Indian reservation or tribe, held in trust by the Federal Government. Trust lands may be held in trust either for a tribe (tribal trust land) or for an individual member of a tribe (individual trust land). Trust lands recognized for the 1990 census comprise all tribal trust lands and inhabited individual trust lands located outside of a reservation boundary. As with other American Indian areas, trust lands may be

located in more than one State. Only the trust lands in a given State are shown in the data products for that State; all trust lands associated with a reservation or tribe are shown in data products for the United States. The Census Bureau first reported data for tribal trust lands for the 1980 census.

Trust lands are assigned a four-digit census code and a five-digit FIPS code, the same as that for the reservation with which they are associated. Trust lands not associated with a reservation are presented by tribal name, interspersed alphabetically among the reservations.

#### Tribal Designated Statistical Area (TDSA)

Tribal designated statistical areas (TDSA's) are areas, delineated outside Oklahoma by federally- and State-recognized tribes without a land base or associated trust lands, to provide statistical areas for which the Census Bureau tabulates data. TDSA's represent areas generally containing the American Indian population over which federally-recognized tribes have jurisdiction and areas in which State tribes provide benefits and services to their members. The names of TDSA's delineated by State-recognized tribes are followed by "(State)." The Census Bureau did not recognize TDSA's before the 1990 census.

Each TDSA is assigned a four-digit census code ranging from 9001 through 9589. The census codes are assigned in alphabetical order of TDSA's nationwide. Each TDSA also is assigned a five-digit FIPS code in alphabetical order within State.

#### Tribal Jurisdiction Statistical Area (TJSA)

Tribal jurisdiction statistical areas (TJSA's) are areas, delineated by federally-recognized tribes in Oklahoma without a reservation, for which the Census Bureau tabulates data. TJSA's represent areas generally containing the American Indian population over which one or more tribal governments have jurisdiction; if tribal officials delineated adjacent TJSA's so that they include some duplicate territory, the overlap area is called a "joint use area," which is treated as a separate TJSA for census purposes.

TJSA's replace the "Historic Areas of Oklahoma (excluding urbanized areas)" shown in 1980 census data products. The Historic Areas of Oklahoma comprised the territory located within reservations that had legally established boundaries from 1900 to 1907; these reservations were dissolved during the 2- to 3-year period preceding the statehood of Oklahoma in 1907. The Historic Areas of Oklahoma (excluding urbanized areas) were identified only for the 1980 census.

Each TJSA is assigned a four-digit census code ranging from 5001 through 5989. The census codes are assigned in alphabetical order of TJSA's, except that joint areas appear at the end of the code range. Each TJSA also is assigned a five-digit FIPS code in alphabetical order within Oklahoma.

#### AREA MEASUREMENT

Area measurements provide the size, in square kilometers (also in square miles in printed reports), recorded for each geographic entity for which the Census Bureau tabulates data in general-purpose data products (except crews-of-vessels entities and ZIP Codes). (Square kilometers may be divided by 2.59 to convert an area measurement to square miles.) Area was calculated from the specific set of boundaries recorded for the entity in the Census Bureau's geographic data base

(see "TIGER"). On machine-readable files, area measurements are shown to three decimal places; the decimal point is implied. In printed reports and listings, area measurements are shown to one decimal.

The Census Bureau provides area measurements for both land and inland water. "Inland water" consists of any lake, reservoir, pond, or similar body of water that is recorded in the Census Bureau's geographic data base. It also includes any river, creek, canal, stream, or similar feature that is recorded in that data base as a two-dimensional feature (rather than as a single line). The portions of the oceans and related large embayments (such as the Chesapeake Bay and Puget Sound), the Gulf of Mexico, and the Caribbean Sea that belong to the United States and its possessions are considered to be "coastal" and "territorial" waters; the Great Lakes are treated as a separate water entity. Rivers and bays that empty into these bodies of water are treated as "inland water" from the point beyond which they are narrower than one nautical mile across. Identification of land and inland, coastal, and territorial waters is for statistical purposes, and does not necessarily reflect legal definitions thereof. By definition, census blocks do not include inland water within their boundaries; therefore, the water area of a block is always zero. Land area measurements may disagree with the information displayed on census maps and in the TIGER file because, for area measurement purposes, features identified as "intermittent water" and "glacier" are reported as land area. For this reason, it may not be possible to derive the land area for an entity by summing the land area of its component census blocks. Crews-of-vessels entities (see "Census Tract and Block Numbering Area" and "Block") do not encompass territory and therefore have no area measurements. ZIP Codes do not have specific boundaries, and therefore, also do not have area measurements.

The accuracy of any area measurement figure is limited by the inaccuracy inherent in (1) the location and shape of the various boundary features in the data base, and (2) rounding affecting the last digit in all operations that compute and/or sum the area measurements.

#### BLOCK

Census blocks are small areas bounded on all sides by visible features such as streets, roads, streams, and railroad tracks, and by invisible boundaries such as city, town, township, and county limits, property lines, and short, imaginary extensions of streets and roads.

Tabulation blocks, used in census data products, are in most cases the same as collection blocks, used in the census enumeration. In some cases, collection blocks have been "split" into two or more parts required for data tabulations. Tabulation blocks do not cross the boundaries of counties, county subdivisions, places, census tracts or block numbering areas, American Indian and Alaska Native areas, congressional districts, voting districts, urban or rural areas, or urbanized areas. The 1990 census is the first for which the entire United States and its possessions are block-numbered.

Blocks are numbered uniquely within each census tract or BNA. A block is identified by a three-digit number, sometimes with a single alphabetical suffix. Block numbers with suffixes generally represent collection blocks that were "split" in order to identify separate geographic entities that divide the original block. For example, when a city limit runs through data collection block 101, the data for the portion inside the city is tabulated in block 101A and the portion outside, in block 101B. A block number with the suffix "Z" represents a "crews-of-vessels" entity for which the Census

Bureau tabulates data, but that does not represent a true geographic area; such a block is shown on census maps associated with an anchor symbol and a census tract or block numbering area with a .99 suffix.

#### BLOCK GROUP (BG)

##### Geographic Block Group

A geographic block group (BG) is a cluster of blocks having the same first digit of their three-digit identifying numbers within a census tract or block numbering area (BNA). For example, BG 3 within a census tract or BNA includes all blocks numbered between 301 and 397. In most cases, the numbering involves substantially fewer than 97 blocks. Geographic BG's never cross census tract or BNA boundaries, but may cross the boundaries of county subdivisions, places, American Indian and Alaska Native areas, urbanized areas, voting districts, and congressional districts. BG's generally contain between 250 and 550 housing units, with the ideal size being 400 housing units.

##### Tabulation Block Group

In the data tabulations, a geographic BG may be split to present data for every unique combination of county subdivision, place, American Indian and Alaska Native area, urbanized area, voting district, urban/rural, and congressional district shown in the data product. BG's are used in tabulating decennial census data nationwide in the 1990 census, in all block-numbered areas in the 1980 census, and in Tape Address Register (TAR) areas in the 1970 census. For purposes of data presentation, BG's are a substitute for the enumeration districts (ED's) used for reporting data in many parts of the United States for the 1970 and 1980 censuses, and in all areas for pre-1970 censuses.

#### BOUNDARY CHANGES

The boundaries of some counties, county subdivisions, American Indian and Alaska Native areas, and many incorporated places, changed between those reported for the 1980 census and January 1, 1990. Boundary changes to legal entities result from:

1. Annexations to or detachments from legally established governmental units.
2. Mergers or consolidations of two or more governmental units.
3. Establishment of new governmental units.
4. Disincorporations or disorganizations of existing governmental units.
5. Changes in treaties and Executive Orders.

The historical counts shown for counties, county subdivisions, and places are not updated for such changes, and thus reflect the population and housing units in the area as delineated at each census. Information on boundary changes reported between the 1980 and 1990 censuses for counties, county subdivisions, and incorporated places is presented in the "User Notes" section of the technical documentation of Summary Tape Files 1 and 3, and in the 1990 CPH-2, Population and Housing Unit Counts printed reports. For information on boundary changes for such areas in the decade preceding other decennial censuses, see the Number of Inhabitants reports for each census. Boundary changes are not reported for some

areas, such as census designated places and block groups.

#### CENSUS TRACT AND BLOCK NUMBERING AREA

##### Block Numbering Area (BNA)

Block numbering areas (BNA's) are small statistical subdivisions of a county for grouping and numbering blocks in nonmetropolitan counties where local census statistical areas committees have not established census tracts. State agencies and the Census Bureau delineated BNA's for the 1990 census, using guidelines similar to those for the delineation of census tracts. BNA's do not cross county boundaries.

BNA's are identified by a four-digit basic number and may have a two-digit suffix; for example, 9901.07. The decimal point separating the four-digit basic BNA number from the two-digit suffix is shown in printed reports, in microfiche, and on census maps; in machine-readable files, the decimal point is implied. Many BNA's do not have a suffix; in such cases, the suffix field is left blank in all data products. BNA numbers range from 9501 through 9989.99, and are unique within a county (numbers in the range of 0001 through 9499.99 denote a census tract). The suffix .99 identifies a BNA that was populated entirely by persons aboard one or more civilian or military ships. A "crews-of-vessels" BNA appears on census maps only as an anchor symbol with its BNA number (and block numbers on maps showing block numbers); the BNA relates to the ships associated with the onshore BNA's having the same four-digit basic number. Suffixes in the range .80 through .98 usually identify BNA's that either were revised or were created during the 1990 census data collection activities. Some of these revisions produced BNA's that have extremely small land area and may have little or no population or housing. For data analysis, such a BNA can be summarized with an adjacent BNA.

##### Census Tract

Census tracts are small, relatively permanent statistical subdivisions of a county. Census tracts are delineated for all metropolitan areas (MA's) and other densely populated counties by local census statistical areas committees following Census Bureau guidelines (more than 3,000 census tracts have been established in 221 counties outside MA's). Six States (California, Connecticut, Delaware, Hawaii, New Jersey, and Rhode Island) and the District of Columbia are covered entirely by census tracts. Census tracts usually have between 2,500 and 8,000 persons and, when first delineated, are designed to be homogeneous with respect to population characteristics, economic status, and living conditions. Census tracts do not cross county boundaries. The spatial size of census tracts varies widely depending on the density of settlement. Census tract boundaries are delineated with the intention of being maintained over a long time so that statistical comparisons can be made from census to census. However, physical changes in street patterns caused by highway construction, new development, etc., may require occasional revisions; census tracts occasionally are split due to large population growth, or combined as a result of substantial population decline. Census tracts are referred to as "tracts" in all 1990 data products.

Census tracts are identified by a four-digit basic number and may have a two-digit suffix; for example, 6059.02. The decimal point separating the four-digit basic tract number from the two-digit suffix is shown in printed reports, in microfiche, and on census maps; in machine-readable files, the decimal point is implied. Many census tracts do not have a suffix; in such cases, the suffix field is left blank in all data products. Leading zeros in a census tract number (for example, 002502) are shown only on machine-readable files.

Census tract numbers range from 0001 through 9499.99 and are unique within a county (numbers in the range of 9501 through 9989.99 denote a block numbering area). The suffix .99 identifies a census tract that was populated entirely by persons aboard one or more civilian or military ships. A "crews-of-vessels" census tract appears on census maps only as an anchor symbol with its census tract number (and block numbers on maps showing block numbers). These census tracts relate to the ships associated with the onshore census tract having the same four-digit basic number. Suffixes in the range .80 through .98 usually identify census tracts that either were revised or were created during the 1990 census data collection activities. Some of these revisions may have resulted in census tracts that have extremely small land area and may have little or no population or housing. For data analysis, such a census tract can be summarized with an adjacent census tract.

#### CONGRESSIONAL DISTRICT

Congressional districts (CD's) are the 435 areas from which persons are elected to the U.S. House of Representatives. After the apportionment of congressional seats among the States, based on census population counts, each State is responsible for establishing CD's for the purpose of electing representatives. Each CD is to be as equal in population to all other CD's in the State as practicable, based on the decennial census counts.

The CD's that were in effect on January 1, 1990 were those of the 101st Congress. Data on the 101st Congress appear in an early 1990 census data product (Summary Tape File 1A). The CD's of the 101st Congress are the same as those in effect for the 102nd Congress. CD's of the 103rd Congress, reflecting redistricting based on the 1990 census, are summarized in later 1990 data products (STF's 1D and 3D, and 1990 CPH-4, Population and Housing Characteristics for Congressional Districts of the 103rd Congress printed reports).

#### COUNTY

The primary political divisions of most States are termed "counties." In Louisiana, these divisions are known as "parishes." In Alaska, which has no counties, the county equivalents are the organized "boroughs" and the "census areas" that are delineated for statistical purposes by the State of Alaska and the Census Bureau. In four States (Maryland, Missouri, Nevada, and Virginia), there are one or more cities that are independent of any county organization and thus constitute primary divisions of their States. These cities are known as "independent cities" and are treated as equivalent to counties for statistical purposes. That part of Yellowstone National Park in Montana is treated as a county equivalent. The District of Columbia has no primary divisions, and the entire area is considered equivalent to a county for statistical purposes.

Each county and county equivalent is assigned a three-digit FIPS code that is unique within State. These codes are assigned in alphabetical order of county or county equivalent within State, except for the independent cities, which follow the listing of counties.

#### COUNTY SUBDIVISION

County subdivisions are the primary subdivisions of counties and their equivalents for the reporting of decennial census data. They include census county divisions, census subareas, minor civil

divisions, and unorganized territories.

Each county subdivision is assigned a three-digit census code in alphabetical order within county and a five-digit FIPS code in alphabetical order within State.

#### Census County Division

Census county divisions (CCD's) are subdivisions of a county that were delineated by the Census Bureau, in cooperation with State officials and local census statistical areas committees, for statistical purposes. CCD's were established in 21 States where there are no legally established minor civil divisions (MCD's), where the MCD's do not have governmental or administrative purposes, where the boundaries of the MCD's change frequently, and/or where the MCD's are not generally known to the public. CCD's have no legal functions, and are not governmental units.

The boundaries of CCD's usually are delineated to follow visible features, and in most cases coincide with census tract or block numbering area boundaries. The name of each CCD is based on a place, county, or well-known local name that identifies its location. CCD's have been established in the following 21 States: Alabama, Arizona, California, Colorado, Delaware, Florida, Georgia, Hawaii, Idaho, Kentucky, Montana, Nevada, New Mexico, Oklahoma, Oregon, South Carolina, Tennessee, Texas, Utah, Washington, and Wyoming. For the 1980 census, the county subdivisions recognized for Nevada were MCD's.

#### Census Subarea (Alaska)

Census subareas are statistical subdivisions of boroughs and census areas (county equivalents) in Alaska. Census subareas were delineated cooperatively by the State of Alaska and the Census Bureau. The census subareas, identified first in 1980, replaced the various types of subdivisions used in the 1970 census.

#### Minor Civil Division

Minor civil divisions (MCD's) are the primary political or administrative divisions of a county. MCD's represent many different kinds of legal entities with a wide variety of governmental and/or administrative functions. MCD's are variously designated as American Indian reservations, assessment districts, boroughs, election districts, gores, grants, magisterial districts, parish governing authority districts, plantations, precincts, purchases, supervisors' districts, towns, and townships. In some States, all or some incorporated places are not located in any MCD and thus serve as MCD's in their own right. In other States, incorporated places are subordinate to (part of) the MCD's in which they are located, or the pattern is mixed--some incorporated places are independent of MCD's and others are subordinate to one or more MCD's.

The Census Bureau recognizes MCD's in the following 28 States: Arkansas, Connecticut, Illinois, Indiana, Iowa, Kansas, Louisiana, Maine, Maryland, Massachusetts, Michigan, Minnesota, Mississippi, Missouri, Nebraska, New Hampshire, New Jersey, New York, North Carolina, North Dakota, Ohio, Pennsylvania, Rhode Island, South Dakota, Vermont, Virginia, West Virginia, and Wisconsin. The District of Columbia has no primary divisions, and the entire area is considered equivalent to an MCD for statistical purposes.

#### Unorganized Territory (unorg.)

In nine States (Arkansas, Iowa, Kansas, Louisiana, Maine, Minnesota, North Carolina, North Dakota, and South Dakota), some counties contain territory that is not included in an MCD recognized by the Census Bureau. Each separate area of unorganized territory in these States is recognized as one or more separate county subdivisions for census purposes. Each unorganized territory is given a descriptive name, followed by the designation "unorg."

#### GEOGRAPHIC CODE

Geographic codes are shown primarily on machine-readable data products, such as computer tape and compact disc-read only memory (CD-ROM), but also appear on other products such as microfiche; they also are shown on some census maps. Codes are identified as "census codes" only if there is also a Federal Information Processing Standards (FIPS) code for the same geographic entity. A code that is not identified as either "census" or "FIPS" is usually a census code for which there is no FIPS equivalent, or for which the Census Bureau does not use the FIPS code. The exceptions, which use only the FIPS code in census products, are county, congressional district, and metropolitan area (that is, metropolitan statistical area, consolidated metropolitan statistical area, and primary metropolitan statistical area).

#### Census Code

Census codes are assigned for a variety of geographic entities, including American Indian and Alaska Native area, census division, census region, county subdivision, place, State, urbanized area, and voting district. The structure, format, and meaning of census codes appear in the 1990 census Geographic Identification Code Scheme; in the technical documentation for summary tape files, CD-ROM's, and microfiche; and in the data dictionaries of summary tape files and CD-ROM's.

#### Federal Information Processing Standards (FIPS) Code

Federal Information Processing Standards (FIPS) codes are assigned for a variety of geographic entities, including American Indian and Alaska Native area, congressional district, county, county subdivision, metropolitan area, place, and State. The structure, format, and meaning of FIPS codes used in the census are shown in the 1990 census Geographic Identification Code Scheme; in the technical documentation for summary tape files, CD-ROM's, and microfiche; and in the data dictionaries of summary tape files and CD-ROM's.

The objective of the FIPS codes is to improve the use of data resources of the Federal Government and avoid unnecessary duplication and incompatibilities in the collection, processing, and dissemination of data. More information about FIPS and FIPS code documentation is available from the National Technical Information Service, Springfield, VA 22161.

#### United States Postal Service (USPS) Code

United States Postal Service (USPS) codes for States are used in all 1990 data products. The codes are two-character alphabetic abbreviations. These codes are the same as the FIPS two-character alphabetic abbreviations.

#### GEOGRAPHIC PRESENTATION

##### Hierarchical Presentation

A hierarchical geographic presentation shows the geographic entities in a superior/subordinate structure in census products. This structure is derived from the legal, administrative, or areal relationships of the entities. The hierarchical structure is depicted in report tables by means of indentation, and is explained for machine-readable media in the discussion of file structure in the technical documentation. An example of hierarchical presentation is the "standard census geographic hierarchy": block, within block group, within census tract or block numbering area, within place, within county subdivision, within county, within State, within division, within region, within the United States. Graphically, this is shown as:

```
United States
  Region
    Division
      State
        County
          County Subdivision
            Place (or part)
              Census tract/block numbering area
                (or part)
                  Block group (or part)
                    Block
```

#### Inventory Presentation

An inventory presentation of geographic entities is one in which all entities of the same type are shown in alphabetical or code sequence, without reference to their hierarchical relationships. Generally, an inventory presentation shows totals for entities that may be split in a hierarchical presentation, such as place, census tract/block numbering area, or block group. An example of a series of inventory presentations is: State, followed by all the counties in that State, followed by all the places in that State. Graphically, this is shown as:

```
State

  County "A"
  County "B"
  County "C"

  Place "X"
  Place "Y"
  Place "Z"
```

#### INTERNAL POINT

An internal point is a set of geographic coordinates (latitude and longitude) that is located within a specified geographic entity. A single point is identified for each entity; for many entities, this point represents the approximate geographic center of that entity. If the shape of the entity caused this point to be located outside the boundaries of the entity, it is relocated from the center so that it is within the entity. If the internal point for a block falls in a water area, it is relocated to a land area within the block. On machine-readable products, internal points are shown to six decimal places; the decimal point is implied.

#### PLACE

Places, for the reporting of decennial census data, include census designated places and incorporated places. Each place is assigned a

four-digit census code that is unique within State. Each place is also assigned a five-digit FIPS code that is unique within State. Both the census and FIPS codes are assigned based on alphabetical order within State. Consolidated cities (see below) are assigned a one-character alphabetical census code that is unique nationwide and a five-digit FIPS code that is unique within State.

#### Census Designated Place (CDP)

Census designated places (CDP's) are delineated for the decennial census as the statistical counterparts of incorporated places. CDP's comprise densely settled concentrations of population that are identifiable by name, but are not legally incorporated places. Their boundaries, which usually coincide with visible features or the boundary of an adjacent incorporated place, have no legal status, nor do these places have officials elected to serve traditional municipal functions. CDP boundaries may change with changes in the settlement pattern; a CDP with the same name as in previous censuses does not necessarily have the same boundaries.

Beginning with the 1950 census, the Census Bureau, in cooperation with State agencies and local census statistical areas committees, has identified and delineated boundaries for CDP's. For the 1990 census, the name of each such place is followed by "CDP." For the 1980 census, "(CDP)" was used; for 1970, 1960, and 1950 censuses, these places were identified by "(U)," meaning "unincorporated place."

To qualify as a CDP for the 1990 census, an unincorporated community must have met the following criteria:

1. In all States except Alaska and Hawaii, the Census Bureau uses three population size criteria to designate a CDP. These criteria are:
  - a. 1,000 or more persons if outside the boundaries of an urbanized area (UA) delineated for the 1980 census or a subsequent special census.
  - b. 2,500 or more persons if inside the boundaries of a UA delineated for the 1980 census or a subsequent special census.
  - c. 250 or more persons if outside the boundaries of a UA delineated for the 1980 census or a subsequent special census, and within the official boundaries of an American Indian reservation recognized for the 1990 census.
2. In Alaska, 25 or more persons if outside a UA, and 2,500 or more persons if inside a UA delineated for the 1980 census or a subsequent special census.
3. In Hawaii, 300 or more persons, regardless of whether the community is inside or outside a UA.

For the 1990 census, CDP's qualified on the basis of the population counts prepared for the 1990 Postcensus Local Review Program. Because these counts were subject to change, a few CDP's may have final population counts lower than the minimums shown above.

Hawaii is the only State with no incorporated places recognized by the Bureau of the Census. All places shown for Hawaii in the data products are CDP's. By agreement with the State of Hawaii, the Census Bureau

does not show data separately for the city of Honolulu, which is coextensive with Honolulu County.

#### Consolidated City

A consolidated government is a unit of local government for which the functions of an incorporated place and its county or minor civil division (MCD) have merged. The legal aspects of this action may result in both the primary incorporated place and the county or MCD continuing to exist as legal entities, even though the county or MCD performs few or no governmental functions and has few or no elected officials. Where this occurs, and where one or more other incorporated places in the county or MCD continue to function as separate governments, even though they have been included in the consolidated government, the primary incorporated place is referred to as a "consolidated city."

The data presentation for consolidated cities varies depending upon the geographic presentation. In hierarchical presentations, consolidated cities are not shown. These presentations include the semi-independent places and the "consolidated city (remainder)." Where the consolidated city is coextensive with a county or county subdivision, the data shown for those areas in hierarchical presentations are equivalent to those for the consolidated government.

For inventory geographic presentations, the consolidated city appears at the end of the listing of places. The data for the consolidated city include places that are part of the consolidated city. The "consolidated city (remainder)" is the portion of the consolidated government minus the semi-independent places, and is shown in alphabetical sequence with other places.

In summary presentations by size of place, the consolidated city is not included. The places semi-independent of consolidated cities are categorized by their size, as is the "consolidated city (remainder)."

Each consolidated city is assigned a one-character alphabetic census code. Each consolidated city also is assigned a five-digit FIPS code that is unique within State. The semi-independent places and the "consolidated city (remainder)" are assigned a four-digit census code and a five-digit FIPS place code that are unique within State. Both the census and FIPS codes are assigned based on alphabetical order within State.

#### Incorporated Place

Incorporated places recognized in 1990 census data products are those reported to the Census Bureau as legally in existence on January 1, 1990 under the laws of their respective States as cities, boroughs, towns, and villages, with the following exceptions: the towns in the New England States, New York, and Wisconsin, and the boroughs in New York are recognized as minor civil divisions for census purposes; the boroughs in Alaska are county equivalents.

#### STATE

States are the primary governmental divisions of the United States. The District of Columbia is treated as a statistical equivalent of a State for census purposes.

Each State and the District of Columbia is assigned a two-digit numeric Federal Information Processing Standards (FIPS) code in alphabetical order by State name. Each State and the District of Columbia also is

assigned a two-digit census code. This code is assigned on the basis of the geographic sequence of each State within each census division; the first digit of the code is the code for the respective division. Each State and equivalent area also is assigned the two-letter FIPS/United States Postal Service (USPS) code.

#### TIGER

TIGER is an acronym for the new digital (computer-readable) geographic data base that automates the mapping and related geographic activities required to support the Census Bureau's census and survey programs. The Census Bureau developed the Topologically Integrated Geographic Encoding and Referencing (TIGER) System to automate the geographic support processes needed to meet the major geographic needs of the 1990 census: producing the cartographic products to support data collection and map publication, providing the geographic structure for tabulation and publication of the collected data, assigning residential and employer addresses to their geographic location and relating those locations to the Census Bureau's geographic units, and so forth. The content of the TIGER data base is made available to the public through a variety of "TIGER Extract" files that may be obtained from the Data User Services Division, U.S. Bureau of the Census, Washington, DC 20233.

#### UNITED STATES

The United States comprises the 50 States and the District of Columbia.

#### VOTING DISTRICT (VTD)

A voting district (VTD) is any of a variety of types of areas (for example, election districts, precincts, wards, legislative districts) established by State and local governments for purposes of elections. For census purposes, each State participating in Phase 2 of the 1990 Census Redistricting Data Program outlined the boundaries of VTD's around groups of whole census blocks on census maps. The entities identified as VTD's are not necessarily those legally or currently established. Also, to meet the "whole block" criterion, a State may have had to adjust VTD boundaries to nearby block boundaries. Therefore, the VTD's shown on the 1990 census tapes, listings, and maps may not represent the actual VTD's in effect at the time of the census. In the 1980 census, VTD's were referred to as "election precincts."

Each VTD is assigned a four-character alphanumeric code that is unique within each county. The code "ZZZZ" is assigned to nonparticipating areas; the Census Bureau reports data for areas coded "ZZZZ."

Census of Population and Housing, 1990: Public Law (P.L.) 94-171 Data on CD-ROM (Name of State) [machine-readable data files] / prepared by the Bureau of the Census. --Washington: The Bureau [producer and distributor], 1991.

## APPENDIX C.

### ACCURACY OF THE DATA

#### CONTENTS

Confidentiality of the Data.....	C-1
Editing of Unacceptable Data.....	C-1
Sources of Error.....	C-1

#### CONFIDENTIALITY OF THE DATA

To maintain confidentiality required by law (Title 13, U.S. Code), the Bureau of the Census applies a confidentiality edit to assure published data do not disclose information about specific individuals, households, and housing units. The result is that a small amount of uncertainty is introduced into some of the census characteristics to prevent identification of specific individuals, households, or housing units. The edit is controlled so that the counts of total persons, totals by race and American Indian tribe, Hispanic origin, and age 18 years and over are not affected by the confidentiality edit and are published as collected. In addition, total counts for housing units by tenure are not affected by this edit.

The confidentiality edit is conducted by selecting a sample of census households from the 100-percent data internal census files and interchanging its data with other households that have identical characteristics on a set of selected key variables but are in different geographic locations within the same State. To provide more protection for "small areas," a higher sampling rate was used for these areas. The net result of this procedure is that the data user's ability to obtain census data, particularly for small areas and subpopulation groups, has been significantly enhanced.

#### EDITING OF UNACCEPTABLE DATA

The objective of the processing operation is to produce a set of data that describes the population as accurately and clearly as possible. To meet this objective, questionnaires were edited during field data collection operations for consistency, completeness, and acceptability. Questionnaires were also reviewed by census clerks for omissions, certain inconsistencies, and population coverage. For example, write-in entries such as "Don't know" or "NA" were considered unacceptable. For some district offices, the initial edit was automated; however, for the majority of the district offices, it was performed by clerks. As a result of this operation, a telephone or personal visit followup was made to obtain missing information. Potential coverage errors were included in the followup, as well as a sample of questionnaires with omissions or inconsistencies.

Subsequent to field operations, remaining incomplete or inconsistent information on the questionnaires was assigned using imputation procedures during the final automated edit of the collected data. Allocations, or computer assignments of acceptable codes in place of unacceptable entries or blanks, are needed most often when an entry for a given item is lacking or when the information reported for a person or housing unit on that item is inconsistent with other information for

that same person or housing unit. As in previous censuses, the general procedure for changing unacceptable entries was to assign an entry for a person or housing unit that was consistent with entries for persons or housing units with similar characteristics. The assignment of acceptable codes in place of blanks or unacceptable entries enhances the usefulness of the data.

Another way in which corrections were made during the computer editing process was through substitution; that is, the assignment of a full set of characteristics for a person or housing unit. When there was an indication that a housing unit was occupied, but the questionnaire contained no information for the people within the household, or the occupants were not listed on the questionnaire, a previously accepted household was selected as a substitute, and the full set of characteristics for the substitute was duplicated. The assignment of the full set of housing characteristics occurred when there was no housing information available. If the housing unit was determined to be occupied, the housing characteristics were assigned from a previously processed occupied unit. If the housing unit was vacant, the housing characteristics were assigned from a previously processed vacant unit.

#### SOURCES OF ERROR

In any large-scale statistical operation, such as the 1990 decennial census, human- and machine-related errors occur. These errors are commonly referred to as nonsampling errors. Such errors include not enumerating every household or every person in the population, not obtaining all required information from the respondents, obtaining incorrect or inconsistent information, and recording information incorrectly. In addition, errors can occur during the field review of the enumerators' work, during clerical handling of the census questionnaires, or during the electronic processing of the questionnaires.

To reduce various types of nonsampling errors, a number of techniques were implemented during the planning, development of the mailing address list, data collection, and data processing activities. Quality assurance methods were used throughout the data collection and processing phases of the census to improve the quality of the data. A reinterview program was designed to minimize the errors in the data collection phase for enumerator-filled questionnaires.

Several coverage improvement programs were implemented during the development of the census address list and census enumeration and processing to minimize undercoverage of the population and housing units. These programs were developed based on experience from the 1980 decennial census and results from the 1990 decennial census testing cycle. In developing and updating the census address list, the Census Bureau used a variety of specialized procedures in different parts of the country.

- In urban areas, the Census Bureau purchased address lists, had the United States Postal Service (USPS) review and update this list, and conducted a dependent canvass and update operation. Prior to mailout, local officials were given the opportunity to examine block counts of address listings (local review) and identify possible errors, and the USPS conducted a final review.
- In more rural parts of the country, the Census Bureau created the address list through a listing operation. The USPS reviewed and updated this list, and the Census Bureau reconciled USPS corrections through a dependent recanvass. Prior to mailout, the USPS conducted a final review, and local officials participated in this review.

Coverage improvement programs continued during and after mailout. The Census Bureau (rather than the USPS) delivered census questionnaires in certain rural areas and in inner city public housing developments. Computer and clerical edits and telephone and personal visit followups also contributed to improved coverage.

If the 1990 census is not subject to count adjustment, the population counts shown in 100-percent data products will be tabulated from the entries for persons on all questionnaires. These counts will not be subject to sampling error. If count adjustment is done, a discussion of the count adjustment methodology and the appropriate methods for calculating sampling errors of adjusted counts can be found in appendix H. (Housing unit counts will not be subject to count adjustment.)

Census of Population and Housing, 1990: Public Law (P.L.) 94-171 Data on CD-ROM (Name of State) [machine-readable data files] / prepared by the Bureau of the Census. --Washington: The Bureau [producer and distributor], 1991.

APPENDIX D.

COLLECTION AND PROCESSING PROCEDURES

CONTENTS

ENUMERATION AND RESIDENCE RULES.....	D-1
Enumeration Rules	
Residence Rules	
Persons in the Armed Forces	
Persons on Maritime Ships	
Persons Away at School	
Persons in Institutions	
Persons Away From Their Usual Residence on Census Data	
DATA COLLECTION PROCEDURES.....	D-2
Enumeration of Housing Units	
Followup	
Special Enumeration Procedures	
Shelter and Street Night (S-Night)	
PROCESSING PROCEDURES.....	D-4

ENUMERATION AND RESIDENCE RULES

In accordance with census practice dating back to the first United States census in 1790, each person was to be enumerated as an inhabitant of his or her "usual residence" in the 1990 census. Usual residence is the place where the person lives and sleeps most of the time or considers to be his or her usual residence. This place is not necessarily the same as the person's legal residence or voting residence. In the vast majority of cases, however, the use of these different bases of classification would produce substantially the same statistics, although there might be appreciable differences for a few areas.

The implementation of this practice has resulted in the establishment of rules for certain categories of persons whose usual place of residence is not immediately apparent. Furthermore, this practice means that persons were not always counted as residents of the place where they happened to be staying on Census Day (April 1, 1990).

Enumeration Rules

Each person whose usual residence was in the United States was to be included in the census, without regard to the person's legal status or citizenship. In a departure from earlier censuses, foreign diplomatic personnel participated voluntarily in the census, regardless of their residence on or off the premises of an embassy. As in previous censuses, persons in the United States specifically excluded from the census were foreign travelers who had not established a residence.

Americans with a usual residence outside the United States were not enumerated in the 1990 census. United States military and Federal civilian employees, and their dependents overseas, are included in the population counts for States for purposes of Congressional apportionment, but are excluded from all other tabulations for States

and their subdivisions. The counts of United States military and Federal civilian employees, and their dependents, were obtained from administrative records maintained by Federal departments and agencies. Other Americans living overseas, such as employees of international agencies and private businesses, and students, were not enumerated, nor were their counts obtained from administrative sources. On the other hand, Americans overseas temporarily were to be enumerated at their usual residence in the United States.

#### Residence Rules

Each person included in the census was to be counted at his or her usual residence--the place where he or she lives and sleeps most of the time or the place where the person considers to be his or her usual home. If a person had no usual residence, the person was to be counted where he or she happened to be staying on April 1, 1990.

Persons temporarily away from their usual residence, whether in the United States or overseas, on a vacation or on a business trip, were counted at their usual residence. Persons who occupied more than one residence during the year were counted at the one they considered to be their usual residence. Persons who moved on or near Census Day were counted at the place they considered to be their usual residence.

Persons in the Armed Forces--Members of the Armed Forces were counted as residents of the area in which the installation was located, either on the installation or in the surrounding community. Family members of Armed Forces personnel were counted where they were living on Census Day (for example, with the Armed Forces person or at another location).

Each Navy ship not deployed to the 6th or the 7th Fleets was attributed to the municipality that the Department of the Navy designated as its homeport. If the homeport included more than one municipality, ships berthed there on Census Day were assigned by the Bureau of the Census to the municipality in which the land immediately adjacent to the dock or pier was actually located. Ships attributed to the homeport, but not physically present and not deployed to the 6th or the 7th Fleets, were assigned to the municipality named on the Department of the Navy's homeport list.

Personnel assigned to each Navy ship were given the opportunity to report a residence off the ship. Those who did report an off-ship residence in the communities surrounding the homeport were counted there; those who did not were counted as residents of the ship. Personnel on Navy ships deployed to the 6th or 7th Fleets on Census Day were considered to be part of the overseas population.

Persons on Maritime Ships--Persons on maritime ships who did not report a usual residence elsewhere were counted as residents of the ship. Those counted on the ship were attributed to:

1. The port where the ship was docked in the United States.
2. The port of departure if between United States ports or between a United States and a foreign port.

If the ship was docked at a foreign port, or between foreign ports, persons counted on the ship were considered to be part of the overseas population.

Persons Away at School--College students were counted as residents of the area in which they were living while attending college, as they have been since the 1950 census. Children in boarding schools below the college level

were counted at their parental home.

Persons in Institutions--Persons under formally authorized, supervised care or custody, such as in Federal or State prisons; local jails; Federal detention centers; juvenile institutions; nursing, convalescent, and rest homes for the aged and dependent; or homes, schools, hospitals, or wards for the physically handicapped, mentally retarded, or mentally ill, were counted at these places.

Persons Away From Their Usual Residence on Census Day--Migrant agricultural workers who did not report a usual residence elsewhere were counted as residents of the place where they were on Census Day. Persons in worker camps who did not report a usual residence elsewhere were counted as residents of the camp where they were on Census Day.

In some parts of the country, natural disasters displaced significant numbers of households from their usual place of residence. If these persons reported a destroyed or damaged residence as their usual residence, they were counted at that location.

Persons away from their usual residence were counted by means of interviews with other members of their families, resident managers, or neighbors.

#### DATA COLLECTION PROCEDURES

The 1990 census was conducted primarily through self-enumeration. Census questionnaires were delivered 1 week before Census Day (April 1, 1990). The questionnaire packet included general information about the 1990 census and an instruction guide explaining how to complete the questionnaire. Spanish-language questionnaires and instruction guides were available on request. Instruction guides also were available in 32 other languages.

#### Enumeration of Housing Units

Each housing unit in the country received one of two versions of the census questionnaire:

1. A short-form questionnaire which contained a limited number of basic population and housing questions; these questions were asked of all persons and housing units and are often referred to as 100-percent questions.
2. A long-form questionnaire which contained the 100-percent items and a number of additional questions; a sampling procedure was used to determine those housing units that were to receive the long-form questionnaire.

Three sampling rates were employed. For slightly more than one-half of the country, one in every six housing units (about 17 percent) received the long-form or sample questionnaire. In functioning local governmental units (counties and incorporated places, and in some parts of the country, towns and townships) estimated to have fewer than 2,500 inhabitants, every other housing unit (50 percent) received the sample questionnaire in order to enhance the reliability of the sample data for these small areas. For census tracts and block numbering areas having more than 2,000 housing units in the Census Bureau's address files, one in every eight housing units (about 13 percent) received a sample questionnaire, providing reliable statistics for these areas while permitting the Census Bureau to stay within a limit of 17.7 million sample questionnaires, or a one-in-six sample, nationwide.

The mail-out/mail-back procedure was used mainly in cities, suburban areas, towns, and rural areas where mailing addresses consisted of a house number and street name. In these areas, the Census Bureau developed mailing lists that included about 88.4 million addresses. The questionnaires were delivered through the mail and respondents were to return them by mail.

The update/leave/mail-back method was used mainly in densely populated rural areas where it was difficult to develop mailing lists because mailing addresses were not house number and street name type. The Census Bureau compiled lists of housing units in advance of the census. The enumerators delivered the questionnaires, asked respondents to return them by mail, and added housing units not on the mailing lists. This method was used mainly in the South, and also included some high-rise, low-income urban areas. A variation of this method was used in urban areas having large numbers of boarded-up buildings. About 11 million housing units were enumerated using this method.

The list/enumerate method (formerly called conventional or door-to-door enumeration) was used mainly in very remote and sparsely-settled areas. The United States Postal Service delivered unaddressed short-form questionnaires before Census Day. Starting a week before Census Day, enumerators canvassed these areas, checked that all housing units received a questionnaire, created a list of all housing units, completed long-form questionnaires, and picked up the completed short-form questionnaires. This method was used mainly in the West and Northeast to enumerate an estimated 6.5 million housing units.

#### Followup

Nonresponse Followup--In areas where respondents were to mail back their questionnaires, an enumerator visited each address from which a questionnaire was not received.

Coverage and Edit-Failure Followup--In the mail-back areas, some households returned a questionnaire that did not meet specific quality standards because of incomplete or inconsistent information, or the respondent had indicated difficulty in deciding who was to be listed on the questionnaire. These households were contacted by telephone or by personal visit to obtain the missing information, or to clarify who was to be enumerated in the household. In areas where an enumerator picked up the questionnaires, the enumerator checked the respondent-filled questionnaire for completeness and consistency.

#### Special Enumeration Procedures

Special procedures and questionnaires were used for the enumeration of persons in group quarters, such as college dormitories, nursing homes, prisons, military barracks, and ships. The questionnaires (Individual Census Reports, Military Census Reports, and Shipboard Census Reports) included the 100-percent population questions, but did not include any housing questions. In all group quarters, all persons were asked the basic population questions; in most group quarters, additional questions were asked of a sample (one-in-six) of persons.

#### Shelter and Street Night (S-Night)

The Census Bureau collected data for various components of the homeless population at different stages in the 1990 census. "Shelter and Street Night" (S-Night) was a special census operation to count the population in four types of locations where homeless people are found. On the evening of March 20, 1990, and during the early morning hours of March 21, 1990, enumerators counted persons in pre-identified

locations:

1. Emergency shelters for the homeless population (public and private; permanent and temporary).
2. Shelters with temporary lodging for runaway youths.
3. Shelters for abused women and their children.
4. Open locations in streets or other places not intended for habitation.

Emergency shelters include all hotels and motels costing \$12 or less (excluding taxes) per night regardless of whether persons living there considered themselves to be homeless, hotels and motels (regardless of cost) used entirely to shelter homeless persons, and pre-identified rooms in hotels and motels used for homeless persons and families. Enumeration in shelters usually occurred from 6 p.m. to midnight; street enumeration, from 2 a.m. to 4 a.m.; abandoned and boarded-up buildings from 4 a.m. to 8 a.m.; and shelters for abused women, from 6 p.m. on March 20 to noon on March 21.

Other components, which some consider as part of the homeless population, were enumerated as part of regular census operations. These include persons doubled up with other families, as well as persons with no other usual home living in transient sites, such as commercial campgrounds, maternity homes for unwed mothers, and drug/alcohol abuse detoxification centers. In institutions, such as local jails and mental hospitals, the Census Bureau does not know who has a usual home elsewhere; therefore, even though some are literally homeless, these persons cannot be identified separately as a component of the homeless population.

There is no generally agreed-upon definition of "the homeless," and there are limitations in the census count that prevent obtaining a total count of the homeless population under any definition. As such, the Census Bureau does not have a definition and will not provide a total count of "the homeless." Rather, the Census Bureau will provide counts and characteristics of persons found at the time of the census in selected types of living arrangements. These selected components can be used as building blocks to construct a count of homeless persons appropriate to particular purposes as long as the data limitations are taken into account.

In preparation for "Shelter-and-Street-Night" enumeration, the regional census centers (RCC's) mailed a certified letter (Form D-33 (L)) to the highest elected official of each active functioning government of the United States (more than 39,000) requesting them to identify:

1. All shelters with sleeping facilities (permanent and temporary, such as church basements, armories, public buildings, and so forth, that could be open on March 20).
2. Hotels and motels used to house homeless persons and families.
3. A list of outdoor locations where homeless persons tend to be at night.
4. Places such as bus or train stations, subway stations, airports, hospital emergency rooms, and so forth, where homeless persons seek shelter at night.

5. The specific addresses of abandoned or boarded-up buildings where homeless persons were thought to stay at night.

The letter from the RCC's to the governmental units emphasized the importance of listing night-time congregating sites. The list of shelters was expanded using information from administrative records and informed local sources. The street sites were limited to the list provided by the jurisdictions. All governmental units were eligible for "Shelter and Street Night." For cities with 50,000 or more persons, the Census Bureau took additional steps to update the list of shelter and street locations if the local jurisdiction did not respond to the certified letter. Smaller cities and rural areas participated if the local jurisdiction provided the Census Bureau a list of shelters or open public places to visit or if shelters were identified through our inventory development, local knowledge update, or during the Special Place Prelist operation.

The Census Bureau encouraged persons familiar with homeless persons and the homeless themselves to apply as enumerators. This recruiting effort was particularly successful in larger cities.

For shelters, both long- and short-form Individual Census Reports (ICR's) were distributed. For street enumeration, only short-form ICR's were used. Persons in shelters and at street locations were asked the basic population questions. Additional questions about social and economic characteristics were asked of a sample of persons in shelters only.

Enumerators were instructed not to ask who was homeless; rather, they were told to count all persons (including children) staying overnight at the shelters, and everyone they saw on the street except the police, other persons in uniform, and persons engaged in employment or obvious money-making activities other than begging and panhandling.

At both shelter and street sites, persons found sleeping were not awakened to answer questions. Rather, the enumerator answered the sex and race questions by observation and estimated the person's age to the best of his or her ability. In shelters, administrative records and information from the shelter operator were used, when available, for persons who were already asleep.

Less than one percent of shelters refused to participate in the census count at first. By the end of the census period, most of those eventually cooperated and the number of refusals had been reduced to a few. For the final refusals, head counts and population characteristics were obtained by enumerators standing outside such shelters and counting people as they left in the morning.

The "street" count was restricted to persons who were visible when the enumerator came to the open, public locations that had been identified by local jurisdictions. Homeless persons who were well hidden, moving about, or in locations other than those identified by the local governments were likely missed. The number missed will never be known and there is no basis to make an estimate of the number missed from census data. The count of persons in open, public places was affected by many factors, including the extra efforts made to encourage people to go to shelters for "Shelter and Street Night," the weather (which was unusually cold in many parts of the country), the presence of the media, and distrust of the census. Expectations of the number of homeless persons on the street cannot be based on the number seen during the day because the night-time situation is normally very different as more homeless persons are in shelters or very well hidden.

For both "Shelter-and-Street-Night" locations, the Census Bureau assumed that the usual home of those enumerated was in the block where they were found (shelter or street).

The "Shelter-and-Street-Night" operation replaced and expanded the 1980 Mission Night (M-Night) and Casual Count operations. These two operations were aimed at counting the population who reported having no usual residence. M-Night was conducted a week after Census Day, in April 1980. Enumerators visited hotels, motels, and similar places costing \$4 or less each night; missions, flophouses, local jails and similar places at which the average length of stay was 30 days or less; and nonshelter locations, such as bus depots, train stations, and all night movie theaters. Questions were asked of everyone, regardless of age. Enumerators conducted M-Night up to midnight on April 8, 1980 and returned the next morning to collect any forms completed after midnight.

The Casual Count operation was conducted in May 1980 at additional nonshelter locations, such as street corners, pool halls, welfare and employment offices. This operation lasted for approximately 2 weeks. Casual Count was conducted during the day only in selected large central cities. Only persons who appeared to be at least 15 years of age were asked if they had been previously enumerated. Casual Count was actually a coverage-improvement operation. It was not specifically an operation to count homeless persons living in the streets. Persons were excluded if they said they had a usual home outside the city because it was not cost effective to check through individual questionnaires in another city to try to find the person.

#### PROCESSING PROCEDURES

Respondents returned many census questionnaires by mail to one of over 344 census district offices or to one of six processing offices. In these offices, the questionnaires were "checked in" and edited for completeness and consistency of the responses. After this initial processing had been performed, all questionnaires were sent to the processing offices.

In the processing offices, the household questionnaires were microfilmed and processed by the Film Optical Sensing Device for Input to Computers (FOSDIC). For most items on the questionnaire, the information supplied by the respondent was indicated by filling circles in predesignated positions. FOSDIC electronically "read" these filled circles from the microfilm copy of the questionnaire and transferred the information to computer disk. The computer tape did not include individual names, addresses, or handwritten responses.

The data processing was performed in several stages. All questionnaires were microfilmed, "read" by FOSDIC, and transferred to computer disk. Selected written entries in the race question on both the short and long forms were keyed from the microfilm and coded using the data base developed from the 1980 census and subsequent content and operational tests. Keying other written entries on the long forms occurred in the seven processing offices.

The information (for example, income dollar amounts or homeowner shelter costs) on these keyed files was merged with the FOSDIC data or processed further through one of three automated coding programs. The codes for industry, occupation, migration, place-of-work, ancestry, language, relationship, race, and Hispanic origin were merged with the FOSDIC data for editing, weighting, and tabulating operations at Census Bureau headquarters. All responses to the questions on Individual

Census Reports (ICR's), Military Census Reports (MCR's), and Shipboard Census Reports (SCR's) were keyed, not processed by microfilm or FOSDIC.

Census of Population and Housing, 1990: Public Law (P.L.) 94-171 Data on CD-ROM (Name of State) [machine-readable data files] / prepared by the Bureau of the Census. --Washington: The Bureau [producer and distributor], 1991.

## DATA DICTIONARY

### CONTENTS

Identification Section	1
Footnote Section	11
Table (Matrix) Section	13

### IDENTIFICATION SECTION

Field name	Data dictionary reference name	Field size	Starting position	Data type
<b>Record Codes</b>				
File Identification(1)	FILEID	8	1	A/N
State/US Abbreviation	STUSAB	2	9	A
AK Alaska	ID Idaho	NC North Carolina	SC South Carolina	
AL Alabama	IL Illinois	ND North Dakota	SD South Dakota	
AR Arkansas	IN Indiana	NE Nebraska	TN Tennessee	
AZ Arizona	KS Kansas	NH New Hampshire	TX Texas	
CA California	KY Kentucky	NJ New Jersey	US United States	
CO Colorado	LA Louisiana	NM New Mexico	UT Utah	
CT Connecticut	MA Massachusetts	NV Nevada	VA Virginia	
DC District of Columbia	MD Maryland	NY New York	VI Virgin Islands	
	ME Maine	OH Ohio	of the United States	
DE Delaware	MI Michigan	OK Oklahoma	VT Vermont	
FL Florida	MN Minnesota	OR Oregon	WA Washington	
GA Georgia	MO Missouri	PA Pennsylvania	WI Wisconsin	
HI Hawaii	MS Mississippi	PR Puerto Rico	WV West Virginia	
IA Iowa	MT Montana	RI Rhode Island	WY Wyoming	
Summary Level(2)	SUMLEV	3	11	N
Geographic Component(3)	GEOCOMP	2	14	N
00 Not a geographic component				
01 Urban				
02 Urban-in urbanized area				
03 Urban-in urbanized area-in urbanized area central place				
04 Urban-in urbanized area-not in urbanized area central place				
05 Urban-not in urbanized area				
06 Urban-not in urbanized area-place [10,000 or more persons]				
07 Urban-not in urbanized area-place [2,500 to 9,999 persons]				
08 Rural				
09 Rural-place [1,000 to 2,499 population, not in an extended city]				
10 Rural-place [0 to 999 population, not in an extended city]				
11 Rural-not in place [or rural part of extended city]				
12 Rural-farm				
13 Urban portion of extended city				
14 Rural portion of extended city				
20 In metropolitan statistical area/consolidated metropolitan statistical area				

21 In metropolitan statistical area/consolidated metropolitan  
     statistical area-urban  
 22 In metropolitan statistical area/consolidated metropolitan  
     statistical area-rural  
 23 In metropolitan statistical area/consolidated metropolitan  
     statistical area-in  
         metropolitan statistical area/primary metropolitan statistical  
         area central city  
 24 In metropolitan statistical area/consolidated metropolitan  
     statistical area-not  
         in metropolitan statistical area/primary metropolitan statistical  
         area central city  
 25 In metropolitan statistical area/consolidated metropolitan  
     statistical area-not  
         in metropolitan statistical area/primary metropolitan statistical  
         area central city-urban  
 26 In metropolitan statistical area/consolidated metropolitan  
     statistical area-not  
         in metropolitan statistical area/primary  
         metropolitan statistical area central city-urban-in urbanized area  
 27 In metropolitan statistical area/consolidated metropolitan  
     statistical area-not  
         in metropolitan statistical area/primary  
         metropolitan statistical area central city-urban-not in urbanized  
         area  
 28 In metropolitan statistical area/consolidated metropolitan  
     statistical area-not  
         in metropolitan statistical area/primary  
         metropolitan statistical area central city-rural  
 30 Not in metropolitan statistical area/consolidated metropolitan  
     statistical area  
 31 Not in metropolitan statistical area/consolidated metropolitan  
     statistical area-  
     urban  
 32 Not in metropolitan statistical area/consolidated metropolitan  
     statistical area-  
     urban-in urbanized area  
 33 Not in metropolitan statistical area/consolidated metropolitan  
     statistical area-  
     urban-not in urbanized area  
 34 Not in metropolitan statistical area/consolidated metropolitan  
     statistical area-  
     urban-not in urbanized area-place [10,000  
     or more population]  
 35 Not in metropolitan statistical area/consolidated metropolitan  
     statistical area-  
     urban-not in urbanized area-place  
     [2,500 - 9,999 population]  
 36 Not in metropolitan statistical area/consolidated metropolitan  
     statistical area-  
     rural  
 40 American Indian reservation and trust land [American Indian  
     reservations (AIR  
     codes 0001-4989) including any trust lands]  
 42 Tribal Jurisdiction Statistical Area [Oklahoma only]  
 43 Tribal Designated Statistical Area  
 44 Alaska Native village statistical area [Alaska only]

Characteristic Iteration(4)	CHARITER	3	16	N
-----------------------------	----------	---	----	---

000 Not a characteristic iteration

Logical Record Number(5)	LOGRECN	6	19	N
--------------------------	---------	---	----	---

Logical Record Part Number(6)	LOGRECPN	4	25	N
Total Number of Parts in Record(7)	PARTREC	4	29	N
Geographic Area Codes				
Alaska Native Regional Corporation(8)	ANRC	2	33	A/N
07 Ahtna	28 Bering Straits	49 Chugach	70 Koniag	
14 Aleut	35 Bristol Bay	56 Cook Inlet	77 NANA	
21 Arctic Slope	42 Calista	63 Doyon	84 Sealaska	
American Indian/Alaska Native Area (Census) (9)	AIANACE	4	35	A/N
American Indian/Alaska Native Area (FIPS) (10)	AIANAFP	5	39	A/N
American Indian/Alaska Native Area Class Code	AIANACC	2	44	A/N
<p>D1 Federally recognized American Indian reservation that does not serve as a minor civil division (MCD) equivalent</p> <p>D2 Federally recognized American Indian reservation that also serves as an MCD equivalent</p> <p>D3 American Indian tribal government that holds property in trust ``trust land'' for a tribe or individual member(s) of the tribe, and the trust land(s) is not associated with a specific American Indian reservation</p> <p>D4 State-recognized American Indian reservation that does not serve as an MCD equivalent</p> <p>D5 State-recognized American Indian reservation that also serves as an MCD equivalent</p> <p>D6 Tribal designated statistical area (TDSA); tribal jurisdiction statistical area (TJSA) (TJSAs occur only in Oklahoma)</p> <p>E1 Alaska Native village statistical area (ANVSA) that does not coincide with or approximate an incorporated place or a census designated place (CDP)</p> <p>E2 ANVSA that coincides with or approximates a CDP</p> <p>E6 ANVSA that coincides with or approximates an incorporated place</p>				
American Indian Reservation Trust Land Indicator Code	ARTLI	1	46	A/N
<p>1 American Indian reservation; including Tribal Jurisdiction Statistical Area (TJSA) and Tribal Designated Statistical Area (TDSA)</p> <p>2 Off-reservation trust land</p> <p>9 Not an American Indian area</p>				
Block(8)	BLCK	4	47	A/N
Block Group(8)	BLCKGR	1	51	A/N
Census Tract/Block Numbering Area(8)	TRACTBNA	6	52	A/N
Congressional District (101st Congress) (8)	CONGDIS	2	58	A/N

01-45 The actual congressional district number

- 00 Applies to States whose representative is elected ``at large'';  
i.e., the State has only one representative  
in the United States House of Representatives
- 98 Applies to areas that have a nonvoting delegate in the United  
States House of Representatives
- 99 Applies to areas that have no representation in the United States  
House of Representatives

Consolidated City (Census) (8) CONCITCE 1 60 A/N

B Butte-Silver Bow, Montana  
C Columbus, Georgia  
I Indianapolis, Indiana  
J Jacksonville, Florida  
M Milford, Connecticut  
N Nashville-Davidson, Tennessee

Consolidated City (FIPS) (8,10) CONCITFP 5 61 A/N

Consolidated City Class Code(10) CONCITCC 2 66 A/N

### C3 Consolidated city

Consolidated City Population Size Code CONCITSC 2 68 A/N

00	Not in universe	12	2,500-4,999
01	0	13	5,000-9,999
02	1-24	14	10,000-19,999
03	25-99	15	20,000-24,999
04	100-199	16	25,000-49,999
05	200-249	17	50,000-99,999
06	250-299	18	100,000-249,999
07	300-499	19	250,000-499,999
08	500-999	20	500,000-999,999
09	1,000-1,499	21	1,000,000-2,499,999
10	1,500-1,999	22	2,500,000-4,999,999
11	2,000-2,499	23	5,000,000 or more

Consolidated Metropolitan Statistical Area (9) CMSA 2 70 A/N

County (9) CNTY 3 72 A/N

County Population Size Code CNTYSC 2 75 A/N

00 Not in universe 12 2,500-4,999

01	0	13	5,000 or less
02	1-24	14	10,000-19,999
03	25-99	15	20,000-24,999
04	100-199	16	25,000-49,999
05	200-249	17	50,000-99,999
06	250-299	18	100,000-249,999
07	300-499	19	250,000-499,999
08	500-999	20	500,000-999,999
09	1,000-1,499	21	1,000,000-2,499,999
10	1,500-1,999	22	2,500,000-4,999,999
11	2,000-2,499	23	5,000,000 or more

County Subdivision (Census) (9) COUSUBCE 3 77 A/N

County Subdivision (FIPS) (10) COUSUBFP 5 80 A/N

County Subdivision Class Code COUSUBCC 2 85 A/N

C2 Incorporated place that also serves as a minor civil division (MCD) equivalent because, although the place is coextensive with an MCD, the Census Bureau, in agreement with State officials, does not recognize that MCD for presenting census data because the MCD cannot provide governmental services (Ohio only)  
 C5 Incorporated place that also serves as an MCD equivalent because it is not part of an MCD or a county subdivision classified as Z5  
 C7 Incorporated place that also serves as a county equivalent and an MCD equivalent; generally referred to as an ``independent city''  
 D2 Federally recognized American Indian reservation that also serves as an MCD equivalent  
 D5 State-recognized American Indian reservation that also serves as an MCD equivalent  
 T1 Governmentally active minor civil division (MCD) that is not coextensive with an incorporated place  
 T5 Governmentally active MCD that is coextensive with an incorporated place  
 Z1 MCD that is governmentally inactive or cannot provide general-purpose governmental services  
 Z3 Unorganized territory identified by the Census Bureau as an MCD equivalent for presenting census data  
 Z5 Census county division (CCD); census subarea (CSA) (Alaska only); census subdistrict (Virgin Islands only)  
 Z7 Incorporated place that the Census Bureau treats as a minor civil division (MCD) equivalent because it is not in any MCD or is coextensive with a legally established but nonfunctioning MCD that the Census Bureau does not recognize for data presentation, AND is located in a State or county whose MCDs cannot provide governmental services (Iowa and Nebraska only)

County Subdivision Population Size Code	COUSUBSC	2	87	A/N
--	----------	---	----	-----

00 Not in universe	12	2,500-4,999
01 0	13	5,000-9,999
02 1-24	14	10,000-19,999
03 25-99	15	20,000-24,999
04 100-199	16	25,000-49,999
05 200-249	17	50,000-99,999
06 250-299	18	100,000-249,999
07 300-499	19	250,000-499,999
08 500-999	20	500,000-999,999
09 1,000-1,499	21	1,000,000-2,499,999
10 1,500-1,999	22	2,500,000-4,999,999
11 2,000-2,499	23	5,000,000 or more

Division(8)	DIVIS	1	89	A/N
-------------	-------	---	----	-----

0 Not in a division	4	West North Central	7	West South Central
1 New England	5	South Atlantic	8	Mountain
2 Middle Atlantic	6	East South Central	9	Pacific
3 East North Central				

Extended City Indicator(8)	EXTCITIN	1	90	A/N
----------------------------	----------	---	----	-----

1 Extended city	9	Not an extended city
-----------------	---	----------------------

Internal Use Code(11)	INTUC	15	91	A/N
-----------------------	-------	----	----	-----

Metropolitan Statistical Area/ Consolidated Metropolitan Statistical Area(9)	MSACMSA	4	106	A/N
MSA/CMSA Population Size Code	MSACMSAS	2	110	A/N
00 Not in universe	12 2,500-4,999			
01 0	13 5,000-9,999			
02 1-24	14 10,000-19,999			
03 25-99	15 20,000-24,999			
04 100-199	16 25,000-49,999			
05 200-249	17 50,000-99,999			
06 250-299	18 100,000-249,999			
07 300-499	19 250,000-499,999			
08 500-999	20 500,000-999,999			
09 1,000-1,499	21 1,000,000-2,499,999			
10 1,500-1,999	22 2,500,000-4,999,999			
11 2,000-2,499	23 5,000,000 or more			
Place (Census) (9)	PLACECE	4	112	A/N
Place (FIPS) (10)	PLACEFP	5	116	A/N
Place Class Code	PLACECC	2	121	A/N
C1 Incorporated place that is governmentally active, is not related to an Alaska Native village statistical area (ANVSA), and does not serve as a minor civil division (MCD) equivalent				
C2 Incorporated place that also serves as a minor civil division (MCD) equivalent because, although the place is coextensive with an MCD, the Census Bureau, in agreement with State officials, does not recognize that MCD for presenting census data because the MCD cannot provide governmental services (Iowa and Ohio only)				
C5 Incorporated place that also serves as an MCD equivalent because it is not part of an MCD				
C6 Incorporated place that coincides with or approximates an ANVSA				
C7 Incorporated place that also serves as a county equivalent; generally referred to as an ``independent city''				
C8 The portion (``remainder'') of a consolidated city that excludes the incorporated place(s) within that jurisdiction				
C9 Incorporated place whose government is operationally inactive and is not included in any other C subclass				
M2 Military or Coast Guard installation (or part of an installation) that serves as a census designated place (CDP)				
U1 CDP with a name that is commonly recognized for the populated area				
U2 CDP with a name that is not commonly recognized for the populated area (e.g., a combination of the names of two or three commonly recognized communities, or a name that identifies the location of the CDP in relation to an adjacent incorporated place)				
U9 CDP that coincides with or approximates an ANVSA.				

Note: In Ohio, a multi-county place that has a different MCD relationship in each county is assigned only a single class code of C1, C2, or C5.

Place Description Code	PLACEDC	1	123	A/N
1 Incorporated central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA), but not a				

central place of an urbanized area (UA)

2 Incorporated central place of an urbanized area (UA), but not a central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA)

3 Incorporated central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA) and a central place of an urbanized area (UA)

4 Consolidated city or an incorporated place that is not a central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA) and not a central place of an urbanized area (UA)

5 Incorporated place, which is the central place of an urbanized area (UA), but only part of which is the central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA)

6 Incorporated place, which is not a central place of an urbanized area (UA), but part of which is the central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA)

A Census designated place (CDP) that is a central place of an urbanized area (UA), but not a central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA)

B Census designated place (CDP) that is a central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA), but not a central place of an urbanized area (UA)

C Census designated place (CDP) that is a central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA) and a central place of an urbanized area (UA)

D Census designated place (CDP) that is in a 1980 or special census urbanized area (UA) and is not a central city or a central place; these CDP's must have a 1990 population of at least 300 in Hawaii and the Virgin Islands of the United States, 1,000 in Puerto Rico, and 2,500 elsewhere

E Census designated place (CDP) not classified elsewhere; these CDP's must meet the following minimum population requirements:

300 in Hawaii and the Virgin Islands of the United States  
25 in Alaska  
1,000 in all other States and Puerto Rico

F Zona urbana that is a central place of an urbanized area (UA) in Puerto Rico, but not a central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA)

G Zona urbana that is a central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA) in Puerto Rico, but not a central place of an urbanized area (UA)

H Zona urbana that is a central city of a metropolitan statistical area/primary metropolitan statistical area (MSA/PMSA) and a central place of an urbanized area (UA)

I Zona urbana in Puerto Rico that is not a central city or a central place

L Census designated place (CDP) entirely within an American Indian reservation and entirely outside of a 1980 or special census urbanized area (UA); these CDP's must have a 1990 population of at least 25 in Alaska and 250 elsewhere

Place	Population	Size	Code	PLACESC	2	124	A/N
00	Not in universe		12	2,500-4,999			
01	0		13	5,000-9,999			
02	1-24		14	10,000-19,999			
03	25-99		15	20,000-24,999			
04	100-199		16	25,000-49,999			
05	200-249		17	50,000-99,999			
06	250-299		18	100,000-249,999			
07	300-499		19	250,000-499,999			
08	500-999		20	500,000-999,999			
09	1,000-1,499		21	1,000,000-2,499,999			
10	1,500-1,999		22	2,500,000-4,999,999			
11	2,000-2,499		23	5,000,000 or more			
Primary Metropolitan Statistical Area(9)				PMSA	4	126	A/N
Region(8)				REG	1	130	A/N
1	Northeast	2	Midwest	3	South	4	West
State (Census) (8)				STATECE	2	131	A/N
06	Puerto Rico			54	Virginia		
07	Virgin Islands of the United States			55	West Virginia		
11	Maine			56	North Carolina		
12	New Hampshire			57	South Carolina		
13	Vermont			58	Georgia		
14	Massachusetts			59	Florida		
15	Rhode Island			61	Kentucky		
16	Connecticut			62	Tennessee		
21	New York			63	Alabama		
22	New Jersey			64	Mississippi		
23	Pennsylvania			71	Arkansas		
31	Ohio			72	Louisiana		
32	Indiana			73	Oklahoma		
33	Illinois			74	Texas		
34	Michigan			81	Montana		
35	Wisconsin			82	Idaho		
41	Minnesota			83	Wyoming		
42	Iowa			84	Colorado		
43	Missouri			85	New Mexico		
44	North Dakota			86	Arizona		
45	South Dakota			87	Utah		
46	Nebraska			88	Nevada		
47	Kansas			91	Washington		
51	Delaware			92	Oregon		
52	Maryland			93	California		
53	District of Columbia			94	Alaska		
				95	Hawaii		
State (FIPS) (8)				STATEFP	2	133	A/N
01	Alabama	17	Illinois	31	Nebraska	46	South Dakota

02	Alaska	18	Indiana	32	Nevada	47	Tennessee
04	Arizona	19	Iowa	33	New Hampshire	48	Texas
05	Arkansas	20	Kansas	34	New Jersey	49	Utah
06	California	21	Kentucky	35	New Mexico	50	Vermont
08	Colorado	22	Louisiana	36	New York	51	Virginia
09	Connecticut	23	Maine	37	North Carolina	53	Washington
10	Delaware	24	Maryland	38	North Dakota	54	West Virginia
11	District of Columbia	25	Massachusetts	39	Ohio	55	Wisconsin
12	Florida	27	Minnesota	41	Oregon	56	Wyoming
13	Georgia	28	Mississippi	42	Pennsylvania	78	Puerto Rico
15	Hawaii	29	Missouri	44	Rhode Island		Virgin Islands of the United States
16	Idaho	30	Montana	45	South Carolina		

Urban/Rural (8)				URBANRUR	1	135	A/N
1	Urban	2	Rural				
Urbanized Area (9)				URBAREA	4	136	A/N
Urbanized Area Population Size Code				UASC	2	140	A/N
00	Not in universe	12	2,500-4,999				
01	0	13	5,000-9,999				
02	1-24	14	10,000-19,999				
03	25-99	15	20,000-24,999				
04	100-199	16	25,000-49,999				
05	200-249	17	50,000-99,999				
06	250-299	18	100,000-249,999				
07	300-499	19	250,000-499,999				
08	500-999	20	500,000-999,999				
09	1,000-1,499	21	1,000,000-2,499,999				
10	1,500-1,999	22	2,500,000-4,999,999				
11	2,000-2,499	23	5,000,000 or more				
Special Area Code [1]				SAC1	5	142	A/N
Special Area Code [2]				SAC2	5	147	A/N
Special Area Code [3]				SAC3	4	152	A/N
Special Area Code [4]				SAC4	4	156	A/N
Special Area Code [5]				SAC5	3	160	A/N
Special Area Code [6]				SAC6	3	163	A/N
Special Area Code [7]				SAC7	2	166	A/N
Special Area Code [8]				SAC8	2	168	A/N
Special Area Code [9]				SAC9	1	170	A/N
Special Area Code [10]				SAC10	1	171	A/N
Area Characteristics							
Area (land) (12)				AREALAND	10	172	A/N
Area (water) (13)				AREAWAT	10	182	A/N

Area Name/PSAD Term/ Part Indicator(14)	ANPSADPI	66	192	A/N
Functional Status Code	FUNCSTAT	1	258	A/N
A Active governmental unit, except American Indian reservations and Alaska Native village statistical areas. An active governmental unit has the legal capacity to have officers, to raise revenue, and to conduct governmental activities under State laws, and currently is doing so.				
B Functioning governmental unit providing no substantive governmental services. These areas may be coextensive with and administered by officials of another governmental unit (such as towns in Connecticut that are coextensive with cities), or the number of officials and/or the functions they perform are so minimal and/or vestigial that the Census Bureau does not recognize them as governmental units for census purposes (such as townships in Iowa).				
F False entity. In order to maintain complete coverage of every State at the county level and of every county at the county subdivision level, we create false entities at these levels for any place that is independent of a county or independent of a minor civil division (MCD). This code also is used for place records that represent the remainder of a consolidated city or the remainder of a county subdivision.				
I Inactive governmental unit. An inactive governmental unit has the legal capacity to be active, but currently has no legal officers, raises no revenues, and conducts no activities.				
N Nonfunctioning governmental unit. A nonfunctioning governmental unit has legally established boundaries, but has no legal capacity to conduct governmental activity.				
R An American Indian reservation, an American Indian tribe whose name is associated with trust lands, or an Alaska Native village statistical area.				
S Statistical entity, except Alaska Native village statistical areas and tribal jurisdiction statistical areas. A statistical entity has no governmental status and is defined by or in cooperation with the Census Bureau or other Federal agency. This category includes census regions, census divisions, census county divisions (CCDs), census designated places (CDPs), metropolitan areas (MSA/CMSA/PMSAs), urbanized areas (UAs), unorganized territories (UTs), tribal designated statistical areas (TDSAs), census areas and census subareas in Alaska, and comunidades and zonas urbanas in Puerto Rico.				
T Tribal jurisdiction statistical area. A tribal jurisdiction statistical area (TJSA) is not a legally defined governmental unit, but is recognized as a statistical area for the 1990 census. These areas exist only in the State of Oklahoma and were not recognized separately in 1980.				

Geographic Change User Note Indicator	GCUNI	1	259	A/N
0 No geographic change note for the area within the summary level				
1 See User Notes for a geographic change note				
Housing Unit Count (100%) (15)	HU100	9	260	A/N
Internal Point (latitude) (16)	INTPTLAT	9	269	A/N
Internal Point (longitude) (17)	INTPTLNG	10	278	A/N
Part Flag	PARTFLAG	1	288	A/N

0 Not a part 1 Part

	Political/Statistical Area Description Code	PSADC	2	289	A/N
01	State or State equivalent - no status is appended to the name of the entity in census publications and related data products.				
04	Borough - county equivalent in Alaska; ``Borough'' is appended to the name of the entity in census publications and related data products.				
05	Census area - county equivalent in Alaska; ``Census Area'' is appended to the name of the entity in census publications and related data products.				
06	County - ``County'' is appended to the name of the entity in census publications and related data products.				
08	Independent city - county equivalent in Maryland, Missouri, and Virginia; ``city'' is appended to the name of the entity in census publications and related data products.				
09	Independent city - county equivalent in Nevada; no status is appended to the name of the entity in census publications and related data products.				
10	Island - county equivalent in Virgin Islands of the United States; ``Island'' is appended to the name of the entity in census publications and related data products.				
13	Municipio - county equivalent in Puerto Rico; ``Municipio'' is appended to the name of the entity in census publications and related data products.				
14	(none) - county equivalent; used at county level for the District of Columbia and Yellowstone National Park (Montana). No status is appended to the name of the entity in census publications and related data products.				
15	Parish - county equivalent in Louisiana; ``Parish'' is appended to the name of the entity in census publications and related data products.				
19	American Indian reservation - minor civil division (MCD) in Maine and New York; ``Reservation'' is appended to the name of the entity in census publications and related data products.				
20	Barrio - minor civil division (MCD) in Puerto Rico; ``barrio'' is appended to the name of the entity in census publications and related data products.				
21	Borough - minor civil division (MCD) in New York; MCD equivalent in New Jersey and Pennsylvania; ``borough'' is appended to the name of the entity in census publications and related data products.				
22	Census county division (CCD) - minor civil division (MCD) equivalent in 21 States; ``division'' is appended to the name of the entity in census publications and related data products.				
23	Census subarea - minor civil division (MCD) equivalent in Alaska; ``census subarea'' is appended to the name of the entity in census publications and related data products.				
24	Census subdistrict - minor civil division (MCD) equivalent in the Virgin Islands of the United States; ``subdistrict'' is appended to the name of the entity in census publications and related data products.				
25	City - minor civil division (MCD) equivalent in 20 States; ``city'' is appended to the name of the entity in census publications and related data products.				
27	District (magisterial, road) - minor civil division (MCD) in Pennsylvania, Virginia, and West Virginia; ``district'' is appended to the name of the entity in census publications and				

related data products.istrict (assessment, election, magisterial, supervisors', or parish governing authority) - minor civil division (MCD) in Louisiana, Maryland, Mississippi, and West Virginia; no status is appended to the name of the entity in census publications and related data products.

29 Election precinct - minor civil division (MCD) in Illinois and Nebraska; ``precinct'' is appended to the name of the entity in census publications and related data products.

30 Election precinct - minor civil division (MCD) in Illinois and Nebraska; no status is appended to the name of the entity in census publications and related data products.

31 Gore - minor civil division (MCD) in Maine and Vermont; ``gore'' is appended to the name of the entity in census publications and related data products.

32 Grant - minor civil division (MCD) in New Hampshire and Vermont; ``grant'' is appended to the name of the entity in census publications and related data products.

33 Independent city - minor civil division (MCD) equivalent in Maryland, Missouri, and Virginia; ``city'' is appended to the name of the entity in census publications and related data products.

34 Independent city - minor civil division (MCD) equivalent in Nevada; no status is appended to the name of the entity in census publications and related data products.

36 Location - minor civil division (MCD) in New Hampshire; ``location'' is appended to the name of the entity in census publications and related data products.

38 (none) - minor civil division (MCD) equivalent for District of Columbia and Arlington County, Virginia; no status is appended to the name of the entity in census publications and related data products.

39 Plantation - minor civil division (MCD) in Maine; ``plantation'' is appended to the name of the entity in census publications and related data products.

40 Plantation - minor civil division (MCD) in Maine; no status is appended to the name of the entity in census publications and related data products.

41 Barrio-pueblo - minor civil division (MCD) in Puerto Rico; ``barrio-pueblo'' is appended to the name of the entity in census publications and related data products.

42 Purchase - minor civil division (MCD) in New Hampshire; ``purchase'' is appended to the name of the entity in census publications and related data products.

43 Town - minor civil division (MCD) in 8 States; MCD equivalent in New Jersey, Pennsylvania, and South Dakota; ``town'' is appended to the name of the entity in census publications and related data products.

44 Township - minor civil division (MCD) in 16 States; ``township'' is appended to the name of the entity in census publications and related data products.

45 Township - minor civil division (MCD) in Kansas, Nebraska, and North Carolina; no status is appended to the name of the entity in census publications and related data products.

46 Unorganized territory - minor civil division (MCD) in 9 States; ``unorg.'' is appended to the name of the entity in census publications and related data products.

47 Village - minor civil division (MCD) equivalent in New Jersey, Ohio, South Dakota, and Wisconsin; ``village'' is appended to the name of the entity in census publications and related data products.

51 Subbarrio - sub-minor civil division (sub-MCD) in Puerto Rico; ``subbarrio'' is appended to the name of the entity in census publications and related data products.

55 Comunidad - place (census designated place) in Puerto Rico; ``comunidad'' is appended to the name of the entity in census publications and related data products.

56 Borough - place in Connecticut, New Jersey, and Pennsylvania; ``borough'' is appended to the name of the entity in census publications and related data products.

57 Census designated place - place; ``CDP'' is appended to the name of the entity in census publications and related data products.

58 City - place; ``city'' is appended to the name of the entity in census publications and related data products.

59 City - place; used for some cities that have a unique description, no description, or a description included with their name, as follows:

- Anaconda-Deer Lodge County (Montana): incorporated municipality;
- Butte-Silver Bow (remainder) (Montana): (none);
- Carson City, Nevada: (none);
- Columbus city (remainder) (Georgia): (none);
- Indianapolis city (remainder) (Indiana): (none);
- Jacksonville city (remainder) (Florida): (none);
- Lexington-Fayette (Kentucky): Urban County Government;
- Lynchburg-Moore County (Tennessee): (none);
- Milford city (remainder) (Connecticut): (none);
- Nashville-Davidson (remainder) (Tennessee): ``Metropolitan Government of Nashville and Davidson County'',

No status is appended to the name of the entity in census publications and related data products.

60 Town - place; place in 30 States and the Virgin Islands of the United States; ``town'' is appended to the name of the entity in census publications and related data products.

61 Village - place; place in 19 States; ``village'' is appended to the name of the entity in census publications and related data products.

62 Zona urbana - place (census designated place) in Puerto Rico; ``zona urbana'' is appended to the name of the entity in census publications and related data products.

65 Consolidated city in Connecticut, Florida, Georgia, and Indiana - ``city'' is appended to the name of the entity in census publications and related data products.

66 Consolidated city - used for some consolidated cities that have unique descriptions or no descriptions, as follows:

- Butte-Silver Bow (Montana): (none);
- Nashville-Davidson (Tennessee): ``Metropolitan Government of Nashville and Davidson County'',

No status is appended to the name of the entity in census publications and related data products.

68 Census region - no status is appended to the name of the entity in census publications and related data products.

69 Census division - no status is appended to the name of the entity in census publications and related data products.

71 Consolidated metropolitan statistical area (CMSA) - ``CMSA'' is appended to the name of the entity in census publications and related data products.

72 Metropolitan statistical area (MSA) - ``MSA'' is appended to the name of the entity in census publications and related data products.

73 Primary metropolitan statistical area (PMSA) - ``PMSA'' is appended to the name of the entity in census publications and related data products.

75 Urbanized area (UA) - no status is appended to the name of the entity in census publications and related data products.

77      Alaska Native Regional Corporation - no status is appended to the name of the entity in census publications and related data products.  
 79      Alaska Native village statistical area - no status is appended to the name of the entity in census publications and related data products.  
 80      Tribal designated statistical area - American Indian reservation equivalent for non-land-based tribes outside of Oklahoma; ``TDSA'' is appended to the name of the entity in census publications and related data products.  
 81      Colony - American Indian reservation; ``Colony'' is appended to the name of the entity in census publications and related data products.  
 82      Community - American Indian reservation; ``Community'' is appended to the name of the entity in census publications and related data products.  
 83      Joint area - American Indian reservation equivalent; ``joint area'' is appended to the name of the entity in census publications and related data products.  
 84      Pueblo - American Indian reservation; ``Pueblo'' is appended to the name of the entity in census publications and related data products.  
 85      Rancheria - American Indian reservation; ``Rancheria'' is appended to the name of the entity in census publications and related data products.  
 86      Reservation - American Indian reservation; ``Reservation'' is appended to the name of the entity in census publications and related data products.  
 87      Reserve - American Indian reservation; ``Reserve'' is appended to the name of the entity in census publications and related data products.  
 88      Tribal jurisdiction statistical area - American Indian reservation equivalent representing historic tribal areas in Oklahoma; ``TJSA'' is appended to the name of the entity in census publications and related data products.  
 89      Trust lands - American Indian reservation equivalent; no status is appended to the name of the entity in census publications and related data products.  
 90      (none) - American Indian reservation; no status is appended to the name of the entity in census publications and related data products.

Population Count (100%) (18)	POP100	9	291	A/N
Special Flag	SPFLAG	1	300	A/N

#### FOOTNOTE SECTION

1 A unique identifier for each file series. Dress rehearsal files have a ``D'' in the character following the last character in the file identification code for the equivalent 1990 decennial census file; for example, PL94171D is the identifier for the dress rehearsal file for the P.L. 94-171 data. File identification code for the P.L. 94-171 file is PL94171. File identification codes for summary tape files are STFnn where nn = file number; for example, STF1A is the identification for Summary Tape File 1A. File identification codes for subject summary tape files are SSTFnn where nn = the number of the subject report. File identification codes for special project files are identified by SPmmm where mmm = a three digit special project number. See How to Use This File for further information.

2. Identifies the geographic level for which the data matrices on the summary tape file have been summarized. The summary level sequence chart describes the hierarchical arrangement of the specified geographic areas with other geographic areas, if any. The summary level must be used in combination with the geographic area codes to identify a specific geographic area (for example, summary level 050 and a specific county code must be used together to locate the data for a particular county). See How To Use This File for further information.

3. Indicates an iteration (repetition), for the specified summary level, of the data matrices on the summary tape file for the geographic components listed in the Geographic Component field. See How To Use This File for further information.

4. Indicates an iteration (repetition), for the specified summary level, of the data matrices on the summary tape file for a population or housing characteristic. Only matrices containing a ``B'' in the prefix have characteristic iterations. See How to Use This File for further information.

5. The logical record is the complete record [identification and set of tables (matrices)] for a geographic entity defined by the summary level, but exclusive of the characteristic iteration. A logical record may have one or more parts (or segments). Each logical record has an assigned sequential integer number within the file. See How to Use This File for further information.

6. Within the logical record, each part is identified uniquely in terms of its sequence. See How to Use This File for further information.

7. This field identifies how many parts (or segments) comprise the entire logical record. See How to Use This File for further information.

8. See appendix A, Area Classifications, for definition of this field if it is applicable to this file.

9. See the publication, Geographic Identification Code Scheme, for codes in this field and related terminology if it is applicable to this file. See appendix A, Area Classifications, for definition of this field if it is applicable to this file.

10. See the publication, Geographic Identification Code Scheme, for FIPS 55 codes in this field and related terminology if it is applicable to this file. See appendix A, Area Classifications, for definition of this field if it is applicable to this file.

11. Codes in unspecified arrangement for Census Bureau use.

12. Land area measurement in thousandths (.001) of a square kilometer. The accuracy of the area measurement is limited by the inaccuracy inherent in 1) the mapping of the various boundary features in the TIGER File and 2) rounding affecting the last digit in all operations that compute and/or sum the area measurements. Land area includes intermittent water and glaciers, which appear on census maps and in the TIGER File as hydrographic features. An area of .0005 square kilometer is rounded to .001; an area smaller than .0005 is rounded to .000. The decimal point is implied on the file. Square miles can be derived by dividing square kilometers by 2.59. See appendix A, Area Classifications, for definition of this field if it is applicable to this file.

13. Water area measurement in thousandths (.001) of a square kilometer. Water area is excluded from census blocks by definition, so the water area

for a block always will be zero. The accuracy of the area measurement is limited by the inaccuracy inherent in 1) the mapping of the various boundary features in the TIGER File and 2) rounding affecting the last digit in all operations that compute and/or sum the area measurements. Water area excludes intermittent water and glaciers, which are treated as land even though they appear on census maps and in the TIGER File as hydrographic features. An area of .0005 square kilometer is rounded to .001; an area smaller than .0005 is rounded to .000. The decimal point is implied on the file. Square miles can be derived by dividing square kilometers by 2.59. See appendix A, Area Classifications, for definition of this field if it is applicable to this file.

14. Name of the lowest-level entity represented by the summary level. In addition to the name of the entity, the name field contains the political/statistical area description (PSAD) when appropriate, and sometimes contains the State abbreviation.

For legal entities, the name is the one reported to the Bureau of the Census in the Boundary and Annexation Survey and by other appropriate sources; for statistical entities, the name is determined by the Office of Management and Budget for metropolitan areas and, for other areas, the name is determined by the Bureau of the Census, usually in cooperation with local officials.

When the summary level represents only part of the area specified in the name, the name usually will have ``(pt.)'' appended to the name/code terminology to designate that this entry for the entity represents only a part of the total entity.

15. The total number of housing units enumerated in the specified summary level as determined in the 100-percent processing.

16. Latitude in degrees, to six decimal places, of a point within the geographic area represented by the summary level. The decimal point is implied on the file. The character immediately preceding the first digit of the latitude of an internal point identifies the direction (hemisphere): a plus sign (+) indicates the Northern Hemisphere; a minus sign (-) indicates the Southern Hemisphere. See appendix A, Area Classifications, for definition of this field if it is applicable to this file.

17. Longitude in degrees, to six decimal places, of a point with the geographic area represented by the summary level. The decimal point is implied on the file. The character immediately preceding the first digit of the longitude of an internal point identifies the direction (hemisphere): a plus sign (+) indicates the Eastern Hemisphere; a minus sign (-) indicates the Western Hemisphere. A point on the 180th meridian is assigned to the Western Hemisphere (-180000000). See appendix A, Area Classifications, for definition of this field if it is applicable to this file.

18. The total number of persons enumerated in the specified summary level as determined in the 100-percent processing.



## APPENDIX F — POPULATION TABLES

This appendix contains the population tables described in section 5.3.2 External Verification and Validation. There are three tables for each licensee. The first table is the SECPOP90 estimated population. The second table is the licensee's reported population. And the third table is a comparison of the SECPOP90 estimated population as a percent of the licensee reported population.

SECPOP90 V2.3 MACCS Site Data File for FitzPatrick - 2, 5 and 10 Mile Radii				
Lat: 43°31'20" Long: 76°24'36" Population multiplier: 1.0000 07-16-1997				
Distance (mi)	2	5	10	Totals
Direction				
N	0	0	0	0
NNE	0	0	0	0
NE	0	0	0	0
ENE	0	0	0	0
E	43	100	653	796
ESE	89	538	2506	3133
SE	48	352	1148	1548
SSE	52	731	955	1738
S	61	574	1332	1967
SSW	101	860	4068	5029
SW	21	547	24661	25229
WSW	3	0	0	3
W	0	0	0	0
WNW	0	0	0	0
NW	0	0	0	0
NNW	0	0	0	0
Totals	418	3702	35323	39443
Licensee's Reported Population - FitzPatrick 3 Radii to 10 Miles				
From Emergency Plan Volume I - Note: Same site as Nine Mile Point				
Distance (mi)	2	5	10	Totals
Direction				
N	0	0	0	0
NNE	0	0	0	0
NE	0	0	0	0
ENE	0	0	0	0
E	63	221	1081	1365
ESE	31	398	2692	3121
SE	115	473	1348	1936
SSE	99	707	1040	1846
S	68	520	1304	1892
SSW	83	1172	4022	5277
SW	66	813	24782	25661
WSW	28	78	3027	3133
W	0	0	0	0
WNW	0	0	0	0
NW	0	0	0	0
NNW	0	0	0	0
Totals	553	4382	39296	44231
SECPOP90 As Percent of Licensee's Reported Population - FitzPatrick 3 Radii to 10 Miles				
Distance (mi)	2	5	10	Totals
Direction				
N				
NNE				
NE				
ENE				
E	68%	45%	60%	58%
ESE	287%	135%	93%	100%
SE	42%	74%	85%	80%
SSE	53%	103%	92%	94%
S	90%	110%	102%	104%
SSW	122%	73%	101%	95%
SW	32%	67%	100%	98%
WSW	11%	*	*	0%
W				
WNW				
NW				
NNW				
Totals	76%	84%	90%	89%

\* The Licensee reported people in the radial segment but SECPOP90 did not locate anyone in the radial segment.

SECPOP90 V2.3 MACCS Site Data File for Millstone 1 - 14 Radii to 50 Miles																	
Lat: 41°18'35" Long: 72°10'10" Population multiplier: 1.0000 07-16-1997																	
Distance (mi)	1	2	3	4	5	6	7	8	9	10	20	30	40	50	Totals		
N	164	707	822	538	145	168	290	417	337	1616	12718	30268	31037	18263	97490		
NNE	64	313	1436	1781	2348	1958	1506	938	1952	3374	45291	24461	31663	30642	147727		
NE	95	582	726	3211	11195	6546	6024	9720	1937	2416	9897	6095	18734	204209	281387		
ENE	154	357	373	4985	1621	7200	5057	2281	1184	2161	33806	15981	41417	96617	213194		
E	0	604	403	1790	198	541	0	11	1255	1199	7933	6689	6584	0	27207		
ESE	0	42	200	0	0	0	0	17	287	24	1	0	836	0	1407		
SE	0	0	0	0	0	0	0	0	0	0	45	1409	0	0	1454		
SSE	0	0	0	0	0	0	0	0	0	0	0	1707	0	0	1707		
S	0	0	0	0	0	0	0	0	0	0	2488	9658	0	0	12146		
SSW	0	0	0	0	0	0	0	0	0	0	2797	11999	19528	179	34503		
SW	0	0	8	0	0	0	0	0	0	0	4499	12703	25377	67416	110003		
WSW	0	52	300	66	82	383	494	92	0	53	1832	0	0	26797	30151		
W	0	224	1167	1039	395	754	508	812	713	866	28340	36820	204426	260338	536402		
WNW	0	541	912	1375	243	281	681	1084	671	328	12126	24691	138491	258079	439503		
NW	0	657	1699	736	1511	495	28	81	71	212	5862	23702	217352	360336	612742		
NNW	0	404	405	1079	772	665	306	1058	233	343	9180	19906	90957	108664	233972		
Totals	477	4483	8451	16600	18510	18991	14894	16511	8640	12592	176815	226089	826402	1431540	2780995		
Licensee's Reported Population - Millstone - 14 Radii to 50 Miles																	
Unit 3's UFSAR Rev. 6, Tables 2.1-4 & 2.1-10																	
Distance (mi)	1	2	3	4	5	6	7	8	9	10	20	30	40	50	Totals		
N	16	722	866	784	116	213	542	209	536	1717	22283	26357	32610	18658	105629		
NNE	13	359	1146	1978	1861	1622	1666	2242	2192	3142	34824	23730	27465	35598	137838		
NE	165	455	839	3888	10584	7752	8164	8129	911	1961	9444	11334	29987	199334	292947		
ENE	22	455	292	4963	971	7186	3748	3047	1008	2662	23914	16498	43001	628307	736074		
E	0	636	413	1804	193	552	0	63	1434	904	10712	7992	10920	0	35623		
ESE	0	143	36	0	0	0	0	0	115	214	0	0	836	0	1344		
SE	0	0	0	0	0	0	0	0	0	0	0	0	807	0	807		
SSE	0	0	0	0	0	0	0	0	0	0	0	0	2420	0	2420		
S	0	0	0	0	0	0	0	0	0	0	0	0	15155	0	15155		
SSW	0	0	0	0	0	0	0	0	0	0	0	0	2443	12569	14807	4498	34317
SW	0	0	14	0	0	0	0	0	0	0	0	0	22980	8252	143933	175179	
WSW	0	0	489	91	86	312	472	158	0	74	2471	0	0	20389	24542		
W	0	178	1061	1014	440	763	475	562	881	408	27956	34384	184723	267465	520310		
WNW	0	476	1165	1964	346	239	211	1654	509	417	12474	27895	148259	259824	455433		
NW	0	634	873	1192	1140	644	599	101	209	81	6215	31331	191767	365578	600364		
NNW	148	314	892	522	646	918	221	429	456	314	8809	17850	115424	78820	225763		
Totals	364	4372	8086	18200	16383	20201	16098	16594	8251	11894	161545	251302	808051	2022404	3363745		
SECPOP90 As Percent of Licensee's Reported Population - Millstone - 14 Radii to 50 Miles																	
Distance (mi)	1	2	3	4	5	6	7	8	9	10	20	30	40	50	Totals		
N	1025%	98%	95%	69%	125%	79%	54%	200%	63%	94%	57%	115%	95%	98%	92%		
NNE	492%	87%	125%	90%	126%	121%	90%	42%	89%	107%	130%	103%	115%	85%	107%		
NE	58%	128%	87%	83%	106%	84%	74%	120%	213%	123%	105%	54%	62%	102%	96%		
ENE	700%	78%	128%	100%	167%	100%	135%	75%	117%	81%	141%	97%	96%	15%	29%		
E	95%	98%	99%	103%	98%			17%	88%	133%	74%	84%	60%		76%		
ESE	29%	556%						**	250%	11%	**	100%			105%		
SE										**	175%				180%		
SSE											71%				71%		
S											**	64%			80%		
SSW											114%	95%	132%	4%	101%		
SW		57%									**	55%	308%	47%	63%		
WSW	**	61%	73%	95%	123%	105%	58%		72%	74%				131%	123%		
W	126%	110%	102%	90%	99%	107%	144%	81%	212%	101%	107%	111%	97%	103%			
WNW	114%	78%	70%	70%	118%	323%	66%	132%	79%	97%	89%	93%	99%	97%			
NW	104%	195%	62%	133%	77%	5%	80%	34%	262%	94%	76%	113%	99%	102%			
NNW	*	129%	45%	207%	120%	72%	138%	247%	51%	109%	104%	112%	79%	138%	104%		
Totals	131%	103%	105%	91%	113%	94%	93%	99%	105%	106%	109%	90%	102%	71%	83%		

\* The Licensee reported people in the radial segment but SECPOP90 did not locate anyone in the radial segment.

\*\* The Licensee did not report anyone in the radial segment but SECPOP90 located people in the radial segment.

SECPOP90 V2.3 MACCS Site Data File for Monticello - 10 Radii to 50 Miles											
Lat: 45°20' 0" Long: 93°50'54" Population multiplier: 1.0000 07-16-1997											
Distance (mi)	1	2	3	4	5	10	20	30	40	50	Totals
Direction											
N	0	0	0	60	100	425	1367	4283	2064	1164	9463
NNE	0	11	9	42	211	668	881	3355	5442	2009	12628
NE	0	0	15	0	101	879	4062	7256	3213	9224	24750
ENE	0	0	20	39	226	1164	4279	5382	13283	10565	34958
E	0	393	83	18	1090	3292	13358	19180	16810	29826	84050
ESE	0	81	78	165	591	2114	11312	122391	206414	234781	577927
SE	0	895	1285	1688	53	908	9984	107073	536360	528544	1186790
SSE	0	158	83	868	538	250	6708	30162	95122	74342	208231
S	3	0	11	0	43	2588	9870	9961	12857	7499	42832
SSW	28	0	195	28	47	1052	3646	6017	5579	8768	25360
SW	0	126	0	30	113	850	3183	5977	7473	10468	28220
WSW	0	0	108	6	303	657	4710	2077	8988	4231	21080
W	0	177	33	40	13	725	2412	3963	3964	5510	16837
WNW	0	56	11	59	89	560	3734	8466	7767	5689	26431
NW	0	0	0	16	14	342	5983	81395	7072	4659	99481
NNW	0	0	0	21	674	830	2294	3018	3921	7076	17834
Totals	31	1897	1931	3080	4206	17304	87783	419956	936329	944355	2416872
Licensee's Reported Population - Monticello 10 Radii to 50 Miles											
Data provided by project manager											
Distance (mi)	1	2	3	4	5	10	20	30	40	50	Totals
Direction											
N	4	58	96	134	171	1421	5071	6872	5590	5036	24453
NNE	8	58	96	134	171	1421	4510	3275	4315	4845	18833
NE	0	54	96	134	171	1367	5629	5277	6916	5460	25104
ENE	0	46	95	136	171	1421	8975	12184	8069	11932	43029
E	0	50	96	134	171	1421	17947	54892	73132	33812	181655
ESE	0	50	96	134	170	1441	43619	114477	190714	299783	650484
SE	0	54	96	137	171	1381	69505	171280	239887	233190	715701
SSE	6	54	96	137	171	1140	41813	157183	91658	36340	328598
S	6	54	96	137	171	1339	5737	19948	18844	15140	61472
SSW	13	54	96	137	171	1436	5737	9211	9382	9304	35541
SW	13	54	96	137	171	1311	5778	7744	6310	7761	29375
WSW	13	54	96	137	171	1360	5400	4054	4531	6223	22039
W	6	54	96	137	171	1360	5090	7206	8345	10319	32784
WNW	0	26	96	137	171	1448	5131	8386	11759	15300	42454
NW	0	42	95	133	171	1421	5393	8124	11461	8351	35191
NNW	0	54	96	134	171	1421	5109	7422	7582	4511	26500
Totals	69	816	1534	2169	2735	22109	240444	597535	698495	707307	2273213
SECPOP90 As Percent of Licensee's Reported Population - Monticello 10 Radii to 50 Miles											
Distance (mi)	1	2	3	4	5	10	20	30	40	50	Totals
Direction											
N	*	*	*	45%	58%	30%	27%	62%	37%	23%	39%
NNE	*	19%	9%	31%	123%	47%	20%	102%	126%	41%	67%
NE	*	*	16%	*	59%	64%	72%	138%	46%	169%	99%
ENE	*	*	21%	29%	132%	82%	48%	44%	165%	89%	81%
E	786%	86%	13%	637%	232%	74%	35%	23%	88%	46%	
ESE	162%	81%	123%	348%	147%	26%	107%	108%	78%	89%	
SE	1657%	1339%	1232%	31%	66%	14%	63%	224%	227%	166%	
SSE	*	293%	86%	634%	315%	22%	16%	19%	104%	205%	63%
S	50%	*	11%	*	25%	193%	172%	50%	68%	50%	70%
SSW	215%	*	203%	20%	27%	73%	64%	65%	59%	94%	71%
SW	*	233%	*	22%	66%	65%	55%	77%	118%	135%	96%
WSW	*	*	113%	4%	177%	48%	87%	51%	198%	68%	96%
W	*	328%	34%	29%	8%	53%	47%	55%	48%	53%	51%
WNW	215%	11%	43%	52%	39%	73%	101%	66%	37%	62%	
NW	*	*	*	12%	8%	24%	111%	1002%	62%	56%	283%
NNW	*	*	*	16%	394%	58%	45%	41%	52%	157%	67%
Totals	45%	232%	126%	142%	154%	78%	37%	70%	134%	134%	106%

\* The Licensee reported people in the radial segment but SECPOP90 did not locate anyone in the radial segment.

SECPOP90 V2.3 MACCS Site Data File for Nine Mile Point 1 & 2 - 25 and 50 Mile Radii							
Lat: 43°31'20" Long: 76°24'36" Population multiplier: 1.0000 07-16-1997							
Distance (mi)	25	50	Totals				
Direction							
N	0	1369	1369				
NNE	829	14333	15162				
NE	3232	68986	72218				
ENE	7025	6628	13653				
E	6126	3713	9839				
ESE	9231	26505	35736				
SE	16992	61328	78320				
SSE	22283	389126	411409				
S	29716	77081	106797				
SSW	14125	52640	66765				
SW	32556	41552	74108				
WSW	3	23456	23459				
W	0	0	0				
WNW	0	0	0				
NW	0	0	0				
NNW	0	0	0				
Totals	142118	766717	908835				
Licensee's Reported Population - Nine Mile Point 50 Mile Radius							
Unit 2's Safety Evaluation Report, 8/7/91, TAC #69095							
Total Population (50 mi radius):		924000					
SECPOP90 As Percent of Licensee's Reported Population - Nine Mile Point 50 Mile Radius							
Total Population (50 mi radius):		98%					

SECPOP90 V2.3 MACCS Site Data File for Oyster Creek									
Lat: 39°48'51" Long: 74°12'23" Population multiplier: 1.0000 07-16-1997									
Distance (mi)	5	10	Totals						
Direction									
N	6172	24689	30861						
NNE	6738	19088	25826						
NE	5124	2039	7163						
ENE	1592	7	1599						
E	821	1	822						
ESE	761	2	763						
SE	1072	1134	2206						
SSE	1446	676	2122						
S	4190	1336	5526						
SSW	3008	9391	12399						
SW	1534	4102	5636						
WSW	12	1222	1234						
W	0	22	22						
WNW	0	10	10						
NW	51	320	371						
NNW	2111	2762	4873						
Totals	34632	66801	101433						
<b>Licensee's Reported Population - Oyster Creek 10 Mile Radius</b>									
Submittal dated 5/19/92									
Total Population (10 mi radius): 96718									
<b>SECPOP90 As Percent of Licensee's Reported Population - Oyster Creek 10 Mile Radius</b>									
Total Population (10 mi radius): 105%									

SECPOP90 V2.3 MACCS Site Data File for Peach Bottom Atomic Power Station - 30 and 60 Mile Radii					
Lat: 39°45'32" Long: 76°16' 9" Population multiplier: 1.0000 07-16-1997					
Distance (mi)	30	60	Totals		
Direction					
N	178888	102442	281330		
NNE	51295	285579	336874		
NE	40551	220399	260950		
ENE	36291	1265357	1301648		
E	70749	713456	784205		
ESE	75281	110240	185521		
SE	21541	85423	106964		
SSE	38454	33304	71758		
S	76583	38287	114870		
SSW	157618	1188166	1345784		
SW	82758	756693	839451		
WSW	19308	124631	143939		
W	31196	116366	147562		
WNW	123248	126270	249518		
NW	49462	353812	403274		
NNW	95456	134540	229996		
Totals	1148679	5654965	6803644		
Licensee's Reported Population - Peach Bottom 60 Mile Radius					
Submittal dated 5/21/92, Accession #9206020160					
Total Population (60 mi radius):		10257315			
SECPOP90 As Percent of Licensee's Reported Population - Peach Bottom 60 Mile Radius					
Total Population (60 mi radius):		66%			

SECPOP90 V2.3 MACCS Site Data File for DC Cook 1 and 2 - 10 Radii to 10 Miles											
Lat: 41°58'34" Long: 86°33'59" Population multiplier: 1.0000 07-16-1997											
Distance (mi)	1	2	3	4	5	6	7	8	9	10	Totals
Direction											
N	0	0	0	0	0	0	0	0	0	0	0
NNE	9	12	62	334	370	1479	1250	1953	2719	5668	13876
NE	1	8	330	1896	2745	3181	2306	3030	1698	4391	19586
ENE	38	139	205	454	207	1024	437	352	673	181	3710
E	0	23	60	204	188	102	168	378	11	410	1544
ESE	0	54	164	407	1155	283	97	251	229	385	3025
SE	0	67	280	84	148	194	81	140	168	145	1307
SSE	12	144	926	404	205	178	177	241	133	106	2526
S	8	76	685	405	231	258	456	392	119	249	2879
SSW	0	2	222	0	29	360	964	317	274	449	2617
SW	0	0	0	0	0	0	0	0	0	0	0
WSW	0	0	0	0	0	0	0	0	0	0	0
W	0	0	0	0	0	0	0	0	0	0	0
WNW	0	0	0	0	0	0	0	0	0	0	0
NW	0	0	0	0	0	0	0	0	0	0	0
NNW	0	0	0	0	0	0	0	0	0	0	0
Totals	68	525	2954	4188	5278	7059	5936	7054	6024	11984	51070
Licensee's Reported Population - DC Cook - 10 Radii to 10 Miles											
Provided by project manager probably from FSAR, 7/93, Figure 2.1-5											
Distance (mi)	1	2	3	4	5	6	7	8	9	10	Totals
Direction											
N	0	0	0	0	0	0	0	0	0	0	0
NNE	16	9	68	325	912	1225	1253	3340	3310	7119	17577
NE	8	28	402	2697	2758	2523	1445	2461	1422	4095	17839
ENE	33	102	188	303	574	472	378	482	489	211	3232
E	7	88	124	158	368	154	172	217	186	512	1986
ESE	0	92	138	437	865	323	184	157	193	454	2843
SE	0	88	217	203	257	172	168	119	129	132	1485
SSE	5	157	777	284	236	178	151	280	102	148	2318
S	5	35	1103	336	234	238	549	234	204	206	3144
SSW	17	67	41	110	13	321	852	486	302	320	2529
SW	0	0	0	0	0	0	0	0	0	0	0
WSW	0	0	0	0	0	0	0	0	0	0	0
W	0	0	0	0	0	0	0	0	0	0	0
WNW	0	0	0	0	0	0	0	0	0	0	0
NW	0	0	0	0	0	0	0	0	0	0	0
NNW	0	0	0	0	0	0	0	0	0	0	0
Totals	91	666	3058	4853	6217	5606	5152	7776	6337	13197	52953
SECPOP90 As Percent of Licensee's Reported Population - DC Cook - 10 Radii to 10 Miles											
Distance (mi)	1	2	3	4	5	6	7	8	9	10	Totals
Direction											
N											
NNE	56%	133%	121%	103%	41%	121%	100%	58%	82%	80%	79%
NE	13%	29%	82%	70%	100%	126%	160%	123%	119%	107%	110%
ENE	115%	136%	109%	150%	36%	217%	116%	73%	138%	86%	115%
E	*	26%	48%	129%	51%	66%	98%	174%	6%	80%	78%
ESE		59%	119%	93%	134%	88%	53%	160%	119%	85%	106%
SE		76%	129%	41%	58%	113%	48%	118%	130%	110%	88%
SSE	240%	92%	119%	142%	87%	100%	117%	86%	130%	72%	109%
S	160%	217%	62%	121%	99%	108%	83%	168%	58%	121%	92%
SSW	*	3%	541%	*	223%	112%	113%	65%	91%	140%	103%
SW											
WSW											
W											
WNW											
NW											
NNW											
Totals	75%	79%	97%	86%	85%	126%	115%	91%	95%	91%	96%

\* The Licensee reported people in the radial segment but SECPOP90 did not locate anyone in the radial segment.

SECPOP90 V2.3 MACCS Site Data File for Diablo Canyon Units 1 and 2 - 10 Radii to 50 Miles												
Lat: 35°12'42" Long: 120°51'16" Population multiplier: 1.0000 07-16-1997												
Distance (mi)	1	2	3	4	5	10	20	30	40	50	Totals	
Direction												
N	0	0	0	0	0	10136	12774	565	2366	333	26174	
NNE	0	0	13	0	0	4968	1990	34237	16989	114	58311	
NE	0	0	0	0	0	228	8909	6648	1005	839	17629	
ENE	0	0	0	0	0	5848	39875	374	5	126	46228	
E	0	0	0	0	0	564	2992	164	16	513	4249	
ESE	0	0	0	0	0	1182	43961	11012	1195	184	57534	
SE	0	0	0	0	0	0	913	46494	52600	3491	103498	
SSE	0	0	0	0	0	0	0	2	9846	45564	55412	
S	0	0	0	0	0	0	0	0	0	0	0	
SSW	0	0	0	0	0	0	0	0	0	0	0	
SW	0	0	0	0	0	0	0	0	0	0	0	
WSW	0	0	0	0	0	0	0	0	0	0	0	
W	0	0	0	0	0	0	0	0	0	0	0	
WNW	0	0	0	0	0	0	0	0	0	0	0	
NW	0	4	0	0	0	0	0	0	44	22	70	
NNW	0	0	0	0	0	4	49	5783	579	492	6907	
Totals	0	4	13	0	0	22930	111463	105279	84645	51678	376012	
Licensee's Reported Population - Diablo Canyon 10 Radii to 50 Miles												
FSAR update												
Distance (mi)	1	2	3	4	5	10	20	30	40	50	Totals	
Direction												
N	0	0	0	0	0	12335	1013	7263	1022	341	21974	
NNE	0	0	0	0	4	1494	11700	1482	6062	1516	22258	
NE	0	0	0	0	0	204	3461	29332	10405	1087	44489	
ENE	0	0	0	0	3	4915	3043	3521	185	1200	12867	
E	0	0	0	0	3	1969	42238	1346	150	1199	46905	
ESE	0	0	0	0	3	1265	7674	323	81	0	9346	
SE	0	0	0	0	0	0	42339	13026	1362	908	57635	
SSE	0	0	0	0	0	0	0	1822	50837	47360	2456	
S	0	0	0	0	0	0	0	0	7010	49738	56748	
SSW	0	0	0	0	0	0	0	0	0	0	0	
SW	0	0	0	0	0	0	0	0	0	0	0	
WSW	0	0	0	0	0	0	0	0	0	0	0	
W	0	0	0	0	0	0	0	0	0	0	0	
WNW	0	0	0	0	0	0	0	0	0	0	0	
NW	0	0	0	0	0	0	0	0	0	0	0	
NNW	0	4	0	0	0	0	0	0	0	0	4	
Totals	0	4	0	0	13	22182	113290	107130	73637	58445	374701	
SECPOP90 As Percent of Licensee's Reported Population - Diablo Canyon 10 Radii to 50 Miles												
Distance (mi)	1	2	3	4	5	10	20	30	40	50	Totals	
Direction												
N						82%	1261%	8%	232%	98%	119%	
NNE		**		*	333%	17%	2310%	280%	8%	262%		
NE					112%	257%	23%	10%	77%	40%		
ENE				*	119%	1310%	11%	3%	11%	359%		
E				*	29%	7%	12%	11%	43%	9%		
ESE				*	93%	573%	3409%	1475%	**	616%		
SE					2%	357%	3862%	384%	180%			
SSE					*	0%	21%	1855%	54%			
S							*	*	*	*		
SSW												
SW												
WSW												
W												
WNW		**						**	**	**		
NW		**					**	**	**	**		
NNW		*				**	**	**	**	**	172675%	
Totals		100%	**	*	103%	98%	98%	115%	88%	100%		

\* The Licensee reported people in the radial segment but SECPOP90 did not locate anyone in the radial segment.

\*\* The Licensee did not report anyone in the radial segment but SECPOP90 located people in the radial segment.

SECPOP90 V2.3 MACCS Site Data File for Fort Calhoun - 5 Radii to 50 Miles						
Lat: 41°31'15" Long: 96°4'36" Population multiplier: 1.0000 07-16-1997						
Distance (mi)	10	20	30	40	50	Totals
<b>Direction</b>						
N	113	390	637	3718	3045	7903
NNE	372	807	833	1455	3362	6829
NE	119	875	2157	2136	2900	8187
ENE	1233	4368	788	4898	4894	16201
E	130	718	2260	3230	2204	8542
ESE	128	1141	2211	3494	1203	8177
SE	416	2396	24506	2555	3547	33420
SSE	1995	149343	143002	16954	3779	315073
S	559	111966	110523	5247	3996	232291
SSW	997	20892	15058	4269	5121	46337
SW	308	2649	5030	4971	3336	16294
WSW	1036	1892	25759	1239	2066	31992
W	367	640	3094	2292	1148	7541
WNW	5804	566	1979	3633	2596	14578
NW	1431	805	2221	4345	2451	11253
NNW	201	2150	536	1336	3245	7468
Totals	15209	301618	340594	65772	48893	772086
<b>Licensee's Reported Population - Fort Calhoun - 5 Radii to 50 Miles</b>						
<b>Radiological Emergency Response Plan, Figure J-7, pg. 25 of 28</b>						
Distance (mi)	10	20	30	40	50	Totals
<b>Direction</b>						
N	36	710	948	3720	1871	7285
NNE	442	499	1067	1578	3351	6937
NE	297	703	2299	2369	1151	6819
ENE	2389	5092	1009	1903	7748	18141
E	116	696	2820	2615	2319	8566
ESE	93	893	1795	3699	2307	8787
SE	886	5432	12296	3151	3012	24777
SSE	1659	107123	151221	13637	3801	277441
S	683	163724	82464	6740	4745	258356
SSW	887	18269	23318	4452	6335	53261
SW	333	2996	4534	5111	3176	16150
WSW	908	2263	22334	1672	1984	29161
W	455	1184	4030	2731	1693	10093
WNW	6123	1085	2060	5494	2838	17600
NW	1076	1271	740	3437	2492	9016
NNW	149	1085	2364	1356	3087	8041
Totals	16532	313025	315299	63665	51910	760431
<b>SECPOP90 As Percent of Licensee's Reported Population - Fort Calhoun - 5 Radii to 50 Miles</b>						
Distance (mi)	10	20	30	40	50	Totals
<b>Direction</b>						
N	314%	55%	67%	100%	163%	108%
NNE	84%	162%	78%	92%	100%	98%
NE	40%	124%	94%	90%	252%	120%
ENE	52%	86%	78%	257%	63%	89%
E	112%	103%	80%	124%	95%	100%
ESE	138%	128%	123%	94%	52%	93%
SE	47%	44%	199%	81%	118%	135%
SSE	120%	139%	95%	124%	99%	114%
S	82%	68%	134%	78%	84%	90%
SSW	112%	114%	65%	96%	81%	87%
SW	92%	68%	111%	97%	105%	101%
WSW	114%	84%	115%	74%	104%	110%
W	81%	54%	77%	84%	68%	75%
WNW	95%	52%	96%	66%	91%	83%
NW	133%	63%	300%	126%	98%	125%
NNW	135%	198%	23%	99%	105%	93%
Totals	92%	96%	108%	103%	94%	102%

SECPOP90 V2.3 MACCS Site Data File for Prairie Island 1 and 2 - 10 Radii to 10 Miles											
Lat: 44°37'19" Long: 92°37'59" Population multiplier: 1.0000 07-16-1997											
Distance (mi)	1	2	3	4	5	6	7	8	9	10	Totals
Direction											
N	0	16	31	47	0	28	23	83	78	97	403
NNE	0	22	50	24	35	28	90	181	151	249	830
NE	0	14	184	15	23	104	182	130	120	171	943
ENE	0	11	14	0	22	37	81	96	80	127	468
E	0	0	6	0	84	53	60	71	67	43	384
ESE	0	0	0	155	168	258	414	179	260	593	2027
SE	0	0	0	2	195	1198	6981	3328	293	139	12136
SSE	14	0	0	733	608	482	205	104	27	95	2268
S	0	0	40	66	747	37	55	50	24	25	1044
SSW	0	0	35	65	23	16	50	121	38	165	513
SW	0	0	30	0	44	67	27	107	22	83	380
WSW	0	10	33	0	36	0	88	26	47	80	320
W	8	32	0	13	89	50	217	24	112	148	693
WNW	113	5	0	60	0	145	442	232	432	116	1545
NW	0	6	7	8	10	9	0	82	328	16	466
NNW	21	0	0	9	31	8	4	0	372	104	549
Totals	156	116	430	1197	2115	2520	8919	4814	2451	2251	24969
Licensee's Reported Population - Prairie Island - 10 Radii to 10 Miles											
Evacuation time estimates, 12/92, Table 3.1											
Distance (mi)	1	2	3	4	5	6	7	8	9	10	Totals
Direction											
N	0	21	61	25	26	27	45	71	114	80	470
NNE	0	54	50	43	48	64	76	124	149	137	745
NE	0	20	31	16	29	86	154	153	186	198	873
ENE	0	11	63	29	37	29	74	73	91	83	490
E	0	2	42	49	42	60	41	61	68	53	418
ESE	0	0	1	56	196	205	733	299	599	429	2518
SE	0	2	0	2	317	1343	6724	3299	220	83	11990
SSE	7	11	4	928	730	158	86	183	57	64	2228
S	0	4	13	244	436	44	34	50	53	53	931
SSW	0	6	31	70	13	15	52	66	98	139	490
SW	0	6	30	17	14	34	100	58	84	66	409
WSW	0	0	35	34	29	33	46	25	50	58	310
W	8	10	3	26	55	133	70	100	128	130	663
WNW	129	54	10	12	32	265	311	437	451	99	1800
NW	22	6	6	2	9	21	1	126	151	163	507
NNW	21	0	0	4	13	21	86	112	108	135	500
Totals	187	207	380	1557	2026	2538	8633	5237	2607	1970	25342
SECPOP90 As Percent of Licensee's Reported Population - Prairie Island - 10 Radii to 10 Miles											
Distance (mi)	1	2	3	4	5	6	7	8	9	10	Totals
Direction											
N		76%	51%	188%	*	104%	51%	117%	68%	121%	86%
NNE		41%	100%	56%	73%	44%	118%	148%	101%	182%	111%
NE		70%	594%	94%	79%	121%	118%	85%	65%	86%	108%
ENE		100%	22%	*	59%	128%	109%	132%	88%	153%	96%
E		*	14%	*	200%	88%	146%	116%	99%	81%	92%
ESE		*	277%	86%	126%	56%	60%	43%	138%	81%	
SE		*	100%	62%	89%	104%	101%	133%	167%	101%	
SSE	200%	*	*	79%	83%	305%	238%	57%	47%	148%	102%
S		*	308%	27%	171%	84%	162%	100%	45%	47%	112%
SSW		*	113%	93%	177%	107%	96%	183%	39%	119%	105%
SW		*	100%	*	314%	197%	27%	184%	26%	126%	93%
WSW		**	94%	*	124%	*	191%	104%	94%	138%	103%
W	100%	320%	*	50%	162%	38%	310%	24%	88%	114%	105%
WNW	88%	9%	*	500%	*	55%	142%	53%	96%	117%	86%
NW	*	100%	117%	400%	111%	43%	*	65%	217%	10%	92%
NNW	100%			225%	238%	38%	5%	*	344%	77%	110%
Totals	83%	56%	113%	77%	104%	99%	103%	92%	94%	114%	99%

\* The Licensee reported people in the radial segment but SECPOP90 did not locate anyone in the radial segment.

\*\* The Licensee did not report anyone in the radial segment but SECPOP90 located people in the radial segment.

SECPOP90 V2.3 MACCS Site Data File for Turkey Point Units 3 and 4 - 10 Radii to 50 Miles											
Lat: 25°26' 6" Long: 80°19'53" Population multiplier: 1.0000 07-16-1997											
Distance (mi)	1	2	3	4	5	10	20	30	40	50	Totals
Direction											
N	0	0	0	0	0	11435	200878	413887	325282	326227	1277709
NNE	0	0	0	0	0	0	10185	429275	352455	195887	987802
NE	0	0	0	0	0	0	0	0	0	0	0
ENE	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	0	6	0	0	0	0	6
ESE	0	0	0	0	0	7	0	0	0	0	7
SE	0	0	0	0	0	0	0	0	0	0	0
SSE	0	0	0	0	0	1265	0	0	0	0	1265
S	0	0	0	0	0	0	1805	228	0	0	2033
SSW	0	0	0	0	3	0	255	9939	6949	1628	18774
SW	0	0	0	0	0	0	0	0	0	12	12
WSW	0	0	0	0	0	57	0	3256	58	195	3566
W	0	0	0	0	0	6407	3982	0	0	0	10389
WNW	0	4	0	0	0	36153	13184	384	0	21	49746
NW	0	0	0	0	0	25338	8204	2	94	18	33656
NNW	0	26	0	0	0	14956	153775	48654	236	222	217869
Totals	0	30	0	0	3	95624	392268	905625	685074	524210	2602834
Licensee's Reported Population - Turkey Point 10 Radii to 50 Miles											
FSAR Section 2.4, Tables 2.4-1 & 2.4-3											
Distance (mi)	1	2	3	4	5	10	20	30	40	50	Totals
Direction											
N	0	0	0	0	0	15799	213226	430335	350347	320863	1330570
NNE	0	0	0	0	0	0	9746	429713	349676	183681	972816
NE	0	0	0	0	0	0	0	0	0	0	0
ENE	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	0	0	0	0	0	0	0
ESE	0	0	0	0	0	0	0	0	0	0	0
SE	0	0	0	0	0	0	0	0	0	0	0
SSE	0	0	0	0	0	1427	0	0	0	0	1427
S	0	0	0	0	0	0	1223	333	0	0	1556
SSW	0	0	0	0	0	0	726	9826	6876	1591	19019
SW	0	0	0	0	0	0	0	0	0	45	45
WSW	0	0	0	0	0	0	0	0	58	190	248
W	0	0	0	0	0	10641	521	0	0	0	11162
WNW	0	0	0	0	0	37006	15205	0	0	23	52234
NW	0	0	0	0	0	24813	8699	0	0	0	33512
NNW	0	0	0	0	0	15993	142481	32254	218	0	190946
Totals	0	0	0	0	0	105679	391827	902461	707175	506393	2613535
SECPOP90 As Percent of Licensee's Reported Population - Turkey Point 10 Radii to 50 Miles											
Distance (mi)	1	2	3	4	5	10	20	30	40	50	Totals
Direction											
N						72%	94%	96%	93%	102%	96%
NNE							105%	100%	101%	107%	102%
NE											
ENE						*					*
E						*					*
ESE						*					*
SE											
SSE						89%					89%
S							148%	68%			131%
SSW						*	35%	101%	101%	102%	99%
SW										27%	27%
WSW						*		*	100%	103%	1438%
W						60%	764%				93%
WNW		*				98%	87%	*		91%	95%
NW						102%	94%	*	*	*	100%
NNW		*				94%	108%	151%	108%	*	114%
Totals	*				*	90%	100%	100%	97%	104%	100%

\* The Licensee did not report anyone in the radial segment but SECPOP90 located people in the radial segment.

## APPENDIX G — CONVERSION PROGRAM LISTINGS

FILENAME: CONVERT1.BAS

```
OPTION EXPLICIT

DECLARE SUB Convert (Filename AS STRING)

' Skip past the DBASE III field information.

CONST CensusFileOffset& = 2946

' Set CR and LF to be the carriage return and line feed characters.

DIM SHARED CR AS STRING * 1, LF AS STRING * 1

CR = CHR$(13)
LF = CHR$(10)

' The structure below will hold the last read census record. The fields
' within the structure hold the information in the fields as described
' in the pb94stru.dbf file.

TYPE PL94171Record

    Flag AS STRING * 1
    Fileid AS STRING * 8
    Stusab AS STRING * 2
    Sumlev AS STRING * 3
    Stuff2 AS STRING * 58
    Cnty AS STRING * 3
    Stuff3 AS STRING * 56
    Statece AS STRING * 2
    Stuff4 AS STRING * 39
    Arealand AS STRING * 10
    Areawat AS STRING * 10
    Anpsadi AS STRING * 66
    Stuff5 AS STRING * 11
    Intplat AS STRING * 9
    Intplon AS STRING * 10
    Stuff6 AS STRING * 3
    Pop100 AS STRING * 9
    Stuff7 AS STRING * 217

END TYPE

OPEN "d:\secpop90\census\county_1.txt" FOR BINARY AS #1
OPEN "d:\secpop90\census\block_1.bin" FOR BINARY AS #2

DIM Dummy AS STRING

PRINT
PRINT "Please insert the AR, IN, LA, MS, SD, WY PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

CALL Convert("e:\pl9417ar.dbf")
CALL Convert("e:\pl9417in.dbf")
CALL Convert("e:\pl9417la.dbf")
CALL Convert("e:\pl9417ms.dbf")
CALL Convert("e:\pl9417sd.dbf")
```

```

CALL Convert("e:\pl9417wy.dbf")

PRINT
PRINT "Please insert the AL, DE, IL, NE, OK, OR PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

CALL Convert("e:\pl9417al.dbf")
CALL Convert("e:\pl9417de.dbf")
CALL Convert("e:\pl9417il.dbf")
CALL Convert("e:\pl9417ne.dbf")
CALL Convert("e:\pl9417ok.dbf")
CALL Convert("e:\pl9417or.dbf")

PRINT
PRINT "Please insert the CO, ID, ME, SC, WA, WV PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

CALL Convert("e:\pl9417co.dbf")
CALL Convert("e:\pl9417id.dbf")
CALL Convert("e:\pl9417me.dbf")
CALL Convert("e:\pl9417sc.dbf")
CALL Convert("e:\pl9417wa.dbf")
CALL Convert("e:\pl9417wv.dbf")

PRINT
PRINT "Please insert the CA, NY PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

CALL Convert("e:\pl9417CA.dbf")
CALL Convert("e:\pl9417NY.dbf")

PRINT
PRINT "Please insert the AZ, GA, MI, NH, ND, WI PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

CALL Convert("e:\pl9417az.dbf")
CALL Convert("e:\pl9417ga.dbf")
CALL Convert("e:\pl9417mi.dbf")
CALL Convert("e:\pl9417nh.dbf")
CALL Convert("e:\pl9417nd.dbf")
CALL Convert("e:\pl9417wi.dbf")

PRINT
PRINT "Please insert the FL, KY, MA, NM, TN, UT PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

CALL Convert("e:\pl9417fl.dbf")
CALL Convert("e:\pl9417ky.dbf")
CALL Convert("e:\pl9417ma.dbf")
CALL Convert("e:\pl9417nm.dbf")
CALL Convert("e:\pl9417tn.dbf")
CALL Convert("e:\pl9417ut.dbf")

PRINT
PRINT "Please insert the MO, MT, NV, TX PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

```

```

CALL Convert("e:\pl9417mo.dbf")
CALL Convert("e:\pl9417mt.dbf")
CALL Convert("e:\pl9417nv.dbf")
CALL Convert("e:\pl9417tx.dbf")

PRINT
PRINT "Please insert the NJ, VT, VA PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

CALL Convert("e:\pl9417nj.dbf")
CALL Convert("e:\pl9417vt.dbf")
CALL Convert("e:\pl9417va.dbf")

PRINT
PRINT "Please insert the CT, DC, MD, NC, OH, RI PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

CALL Convert("e:\pl9417ct.dbf")
CALL Convert("e:\pl9417dc.dbf")
CALL Convert("e:\pl9417md.dbf")
CALL Convert("e:\pl9417nc.dbf")
CALL Convert("e:\pl9417oh.dbf")
CALL Convert("e:\pl9417ri.dbf")

PRINT
PRINT "Please insert the IA, KS, MN, PA PL94-171 CD-ROM."
INPUT ; "Press Return to Continue...", Dummy
PRINT : PRINT

CALL Convert("e:\pl9417ia.dbf")
CALL Convert("e:\pl9417ks.dbf")
CALL Convert("e:\pl9417mn.dbf")
CALL Convert("e:\pl9417pa.dbf")

CLOSE #1, #2

END

SUB Convert (Filename AS STRING)

    DIM PL94171 AS PL94171Record
    DIM Longitude AS LONG, Latitude AS LONG
    DIM Population AS LONG, Area AS LONG
    DIM State AS STRING * 1, County AS INTEGER

    OPEN Filename FOR BINARY AS #3

    SEEK #3, CensusFileOffset&

    DO

        GET #3, , PL94171

        IF EOF(3) THEN

            EXIT DO

        ELSEIF (PL94171.Sumlev = "040") THEN

```

```

PRINT PL94171.Stusab; " "; PL94171.Statece; " "; PL94171.Anpsadi
LPRINT PL94171.Stusab; " "; PL94171.Statece; " "; PL94171.Anpsadi

ELSEIF (PL94171.Sumlev = "050") THEN

PRINT PL94171.Stusab; " "; PL94171.Cnty; " "; PL94171.Anpsadi
LPRINT PL94171.Stusab; " "; PL94171.Cnty; " "; PL94171.Anpsadi

PUT #1, , PL94171.Stusab
PUT #1, , PL94171.Statece
PUT #1, , PL94171.Cnty
PUT #1, , PL94171.Anpsadi
PUT #1, , PL94171.Pop100
PUT #1, , PL94171.Arealand
PUT #1, , PL94171.Areawat
PUT #1, , CR
PUT #1, , LF

ELSEIF (PL94171.Sumlev = "750") THEN

Longitude = -VAL(PL94171.Intplon)
Latitude = VAL(PL94171.Intplat)
Population = VAL(PL94171.Pop100)
Area = VAL(PL94171.Arealand)
State = CHR$(VAL(PL94171.Statece))
County = VAL(PL94171.Cnty)

PUT #2, , Longitude
PUT #2, , Latitude
PUT #2, , Population
PUT #2, , Area
PUT #2, , State
PUT #2, , County

END IF

LOOP

CLOSE #3

END SUB

```

**FILENAME: CONVERT2.BAS**

```
DECLARE SUB MakeCountyConvert ()
OPTION EXPLICIT

CONST MAXSTATE = 93
CONST MAXCOUNTY = 840

DIM Longitude AS LONG, Latitude AS LONG
DIM Population AS LONG, Area AS LONG
DIM State AS STRING * 1, County AS INTEGER
DIM PresentState AS STRING * 1, PresentCounty AS INTEGER

DIM ShortLongitude AS INTEGER, ShortLatitude AS INTEGER
DIM ShortPopulation AS INTEGER, LongArea AS LONG
DIM ShortCounty AS INTEGER

REDIM SHARED CountyConvert(MAXSTATE, MAXCOUNTY) AS INTEGER

CALL MakeCountyConvert

OPEN "d:\secpop90\census\block_1.bin" FOR BINARY AS #1
OPEN "d:\secpop90\census\block_2.bin" FOR BINARY AS #2

PresentState = CHR$(0)
PresentCounty = 0

DO

    GET #1, , Longitude
    GET #1, , Latitude
    GET #1, , Population
    GET #1, , Area
    GET #1, , State
    GET #1, , County

    IF EOF(1) THEN
        EXIT DO
    ELSE
        ShortLongitude = CINT((Longitude / 1000) - 91993)
        ShortLatitude = CINT((Latitude / 1000) - 16610)
        ShortPopulation = INT(Population)
        LongArea = Area
        ShortCounty = CountyConvert(ASC(State), County)

        IF ((PresentState <> State) OR (PresentCounty <> County)) THEN
            PresentState = State
            PresentCounty = County
            PRINT ASC(PresentState), PresentCounty, ShortCounty
            LPRINT ASC(PresentState), PresentCounty, ShortCounty
        END IF

        IF (ShortCounty = 0) THEN
            PRINT "County converter error, ShortCounty = 0!"
            EXIT DO
        END IF

        PUT #2, , ShortLongitude
        PUT #2, , ShortLatitude
    END IF
END DO
```

```

    PUT #2, , ShortPopulation
    PUT #2, , LongArea
    PUT #2, , ShortCounty

    END IF

    LOOP

CLOSE #1, #2

END

'Data for converting census/FIPS county designations to a single integer.
'The format is Census State Number, FIPS County Number, SECPOP90 County Number.
'There are ten counties per line except for the last line which only has one.

DATA 11, 1, 1, 11, 3, 2, 11, 5, 3, 11, 7, 4, 11, 9, 5, 11, 11, 6, 11, 13, 7, 11, 15,
8, 11, 17, 9, 11, 19, 10
DATA 11, 21, 11, 11, 23, 12, 11, 25, 13, 11, 27, 14, 11, 29, 15, 11, 31, 16, 12, 1,
17, 12, 3, 18, 12, 5, 19, 12, 7, 20
DATA 12, 9, 21, 12, 11, 22, 12, 13, 23, 12, 15, 24, 12, 17, 25, 12, 19, 26, 13, 1,
27, 13, 3, 28, 13, 5, 29, 13, 7, 30
DATA 13, 9, 31, 13, 11, 32, 13, 13, 33, 13, 15, 34, 13, 17, 35, 13, 19, 36, 13, 21,
37, 13, 23, 38, 13, 25, 39, 13, 27, 40
DATA 14, 1, 41, 14, 3, 42, 14, 5, 43, 14, 7, 44, 14, 9, 45, 14, 11, 46, 14, 13, 47,
14, 15, 48, 14, 17, 49, 14, 19, 50
DATA 14, 21, 51, 14, 23, 52, 14, 25, 53, 14, 27, 54, 15, 1, 55, 15, 3, 56, 15, 5,
57, 15, 7, 58, 15, 9, 59, 16, 1, 60
DATA 16, 3, 61, 16, 5, 62, 16, 7, 63, 16, 9, 64, 16, 11, 65, 16, 13, 66, 16, 15, 67,
21, 1, 68, 21, 3, 69, 21, 5, 70
DATA 21, 7, 71, 21, 9, 72, 21, 11, 73, 21, 13, 74, 21, 15, 75, 21, 17, 76, 21, 19,
77, 21, 21, 78, 21, 23, 79, 21, 25, 80
DATA 21, 27, 81, 21, 29, 82, 21, 31, 83, 21, 33, 84, 21, 35, 85, 21, 37, 86, 21, 39,
87, 21, 41, 88, 21, 43, 89, 21, 45, 90
DATA 21, 47, 91, 21, 49, 92, 21, 51, 93, 21, 53, 94, 21, 55, 95, 21, 57, 96, 21, 59,
97, 21, 61, 98, 21, 63, 99, 21, 65, 100
DATA 21, 67, 101, 21, 69, 102, 21, 71, 103, 21, 73, 104, 21, 75, 105, 21, 77, 106,
21, 79, 107, 21, 81, 108, 21, 83, 109, 21, 85, 110
DATA 21, 87, 111, 21, 89, 112, 21, 91, 113, 21, 93, 114, 21, 95, 115, 21, 97, 116,
21, 99, 117, 21, 101, 118, 21, 103, 119, 21, 105, 120
DATA 21, 107, 121, 21, 109, 122, 21, 111, 123, 21, 113, 124, 21, 115, 125, 21, 117,
126, 21, 119, 127, 21, 121, 128, 21, 123, 129, 22, 1, 130
DATA 22, 3, 131, 22, 5, 132, 22, 7, 133, 22, 9, 134, 22, 11, 135, 22, 13, 136, 22,
15, 137, 22, 17, 138, 22, 19, 139, 22, 21, 140
DATA 22, 23, 141, 22, 25, 142, 22, 27, 143, 22, 29, 144, 22, 31, 145, 22, 33, 146,
22, 35, 147, 22, 37, 148, 22, 39, 149, 22, 41, 150
DATA 23, 1, 151, 23, 3, 152, 23, 5, 153, 23, 7, 154, 23, 9, 155, 23, 11, 156, 23,
13, 157, 23, 15, 158, 23, 17, 159, 23, 19, 160
DATA 23, 21, 161, 23, 23, 162, 23, 25, 163, 23, 27, 164, 23, 29, 165, 23, 31, 166,
23, 33, 167, 23, 35, 168, 23, 37, 169, 23, 39, 170
DATA 23, 41, 171, 23, 43, 172, 23, 45, 173, 23, 47, 174, 23, 49, 175, 23, 51, 176,
23, 53, 177, 23, 55, 178, 23, 57, 179, 23, 59, 180
DATA 23, 61, 181, 23, 63, 182, 23, 65, 183, 23, 67, 184, 23, 69, 185, 23, 71, 186,
23, 73, 187, 23, 75, 188, 23, 77, 189, 23, 79, 190
DATA 23, 81, 191, 23, 83, 192, 23, 85, 193, 23, 87, 194, 23, 89, 195, 23, 91, 196,
23, 93, 197, 23, 95, 198, 23, 97, 199, 23, 99, 200
DATA 23, 101, 201, 23, 103, 202, 23, 105, 203, 23, 107, 204, 23, 109, 205, 23, 111,
206, 23, 113, 207, 23, 115, 208, 23, 117, 209, 23, 119, 210
DATA 23, 121, 211, 23, 123, 212, 23, 125, 213, 23, 127, 214, 23, 129, 215, 23, 131,
216, 23, 133, 217, 31, 1, 218, 31, 3, 219, 31, 5, 220
DATA 31, 7, 221, 31, 9, 222, 31, 11, 223, 31, 13, 224, 31, 15, 225, 31, 17, 226, 31,
19, 227, 31, 21, 228, 31, 23, 229, 31, 25, 230

```

DATA 31, 27, 231, 31, 29, 232, 31, 31, 233, 31, 33, 234, 31, 35, 235, 31, 37, 236, 31, 39, 237, 31, 41, 238, 31, 43, 239, 31, 45, 240  
 DATA 31, 47, 241, 31, 49, 242, 31, 51, 243, 31, 53, 244, 31, 55, 245, 31, 57, 246, 31, 59, 247, 31, 61, 248, 31, 63, 249, 31, 65, 250  
 DATA 31, 67, 251, 31, 69, 252, 31, 71, 253, 31, 73, 254, 31, 75, 255, 31, 77, 256, 31, 79, 257, 31, 81, 258, 31, 83, 259, 31, 85, 260  
 DATA 31, 87, 261, 31, 89, 262, 31, 91, 263, 31, 93, 264, 31, 95, 265, 31, 97, 266, 31, 99, 267, 31, 101, 268, 31, 103, 269, 31, 105, 270  
 DATA 31, 107, 271, 31, 109, 272, 31, 111, 273, 31, 113, 274, 31, 115, 275, 31, 117, 276, 31, 119, 277, 31, 121, 278, 31, 123, 279, 31, 125, 280  
 DATA 31, 127, 281, 31, 129, 282, 31, 131, 283, 31, 133, 284, 31, 135, 285, 31, 137, 286, 31, 139, 287, 31, 141, 288, 31, 143, 289, 31, 145, 290  
 DATA 31, 147, 291, 31, 149, 292, 31, 151, 293, 31, 153, 294, 31, 155, 295, 31, 157, 296, 31, 159, 297, 31, 161, 298, 31, 163, 299, 31, 165, 300  
 DATA 31, 167, 301, 31, 169, 302, 31, 171, 303, 31, 173, 304, 31, 175, 305, 32, 1, 306, 32, 3, 307, 32, 5, 308, 32, 7, 309, 32, 9, 310  
 DATA 32, 11, 311, 32, 13, 312, 32, 15, 313, 32, 17, 314, 32, 19, 315, 32, 21, 316, 32, 23, 317, 32, 25, 318, 32, 27, 319, 32, 29, 320  
 DATA 32, 31, 321, 32, 33, 322, 32, 35, 323, 32, 37, 324, 32, 39, 325, 32, 41, 326, 32, 43, 327, 32, 45, 328, 32, 47, 329, 32, 49, 330  
 DATA 32, 51, 331, 32, 53, 332, 32, 55, 333, 32, 57, 334, 32, 59, 335, 32, 61, 336, 32, 63, 337, 32, 65, 338, 32, 67, 339, 32, 69, 340  
 DATA 32, 71, 341, 32, 73, 342, 32, 75, 343, 32, 77, 344, 32, 79, 345, 32, 81, 346, 32, 83, 347, 32, 85, 348, 32, 87, 349, 32, 89, 350  
 DATA 32, 91, 351, 32, 93, 352, 32, 95, 353, 32, 97, 354, 32, 99, 355, 32, 101, 356, 32, 103, 357, 32, 105, 358, 32, 107, 359, 32, 109, 360  
 DATA 32, 111, 361, 32, 113, 362, 32, 115, 363, 32, 117, 364, 32, 119, 365, 32, 121, 366, 32, 123, 367, 32, 125, 368, 32, 127, 369, 32, 129, 370  
 DATA 32, 131, 371, 32, 133, 372, 32, 135, 373, 32, 137, 374, 32, 139, 375, 32, 141, 376, 32, 143, 377, 32, 145, 378, 32, 147, 379, 32, 149, 380  
 DATA 32, 151, 381, 32, 153, 382, 32, 155, 383, 32, 157, 384, 32, 159, 385, 32, 161, 386, 32, 163, 387, 32, 165, 388, 32, 167, 389, 32, 169, 390  
 DATA 32, 171, 391, 32, 173, 392, 32, 175, 393, 32, 177, 394, 32, 179, 395, 32, 181, 396, 32, 183, 397, 33, 1, 398, 33, 3, 399, 33, 5, 400  
 DATA 33, 7, 401, 33, 9, 402, 33, 11, 403, 33, 13, 404, 33, 15, 405, 33, 17, 406, 33, 19, 407, 33, 21, 408, 33, 23, 409, 33, 25, 410  
 DATA 33, 27, 411, 33, 29, 412, 33, 31, 413, 33, 33, 414, 33, 35, 415, 33, 37, 416, 33, 39, 417, 33, 41, 418, 33, 43, 419, 33, 45, 420  
 DATA 33, 47, 421, 33, 49, 422, 33, 51, 423, 33, 53, 424, 33, 55, 425, 33, 57, 426, 33, 59, 427, 33, 61, 428, 33, 63, 429, 33, 65, 430  
 DATA 33, 67, 431, 33, 69, 432, 33, 71, 433, 33, 73, 434, 33, 75, 435, 33, 77, 436, 33, 79, 437, 33, 81, 438, 33, 83, 439, 33, 85, 440  
 DATA 33, 87, 441, 33, 89, 442, 33, 91, 443, 33, 93, 444, 33, 95, 445, 33, 97, 446, 33, 99, 447, 33, 101, 448, 33, 103, 449, 33, 105, 450  
 DATA 33, 107, 451, 33, 109, 452, 33, 111, 453, 33, 113, 454, 33, 115, 455, 33, 117, 456, 33, 119, 457, 33, 121, 458, 33, 123, 459, 33, 125, 460  
 DATA 33, 127, 461, 33, 129, 462, 33, 131, 463, 33, 133, 464, 33, 135, 465, 33, 137, 466, 33, 139, 467, 33, 141, 468, 33, 143, 469, 33, 145, 470  
 DATA 33, 147, 471, 33, 149, 472, 33, 151, 473, 33, 153, 474, 33, 155, 475, 33, 157, 476, 33, 159, 477, 33, 161, 478, 33, 163, 479, 33, 165, 480  
 DATA 33, 167, 481, 33, 169, 482, 33, 171, 483, 33, 173, 484, 33, 175, 485, 33, 177, 486, 33, 179, 487, 33, 181, 488, 33, 183, 489, 33, 185, 490  
 DATA 33, 187, 491, 33, 189, 492, 33, 191, 493, 33, 193, 494, 33, 195, 495, 33, 197, 496, 33, 199, 497, 33, 201, 498, 33, 203, 499, 34, 1, 500  
 DATA 34, 3, 501, 34, 5, 502, 34, 7, 503, 34, 9, 504, 34, 11, 505, 34, 13, 506, 34, 15, 507, 34, 17, 508, 34, 19, 509, 34, 21, 510  
 DATA 34, 23, 511, 34, 25, 512, 34, 27, 513, 34, 29, 514, 34, 31, 515, 34, 33, 516, 34, 35, 517, 34, 37, 518, 34, 39, 519, 34, 41, 520  
 DATA 34, 43, 521, 34, 45, 522, 34, 47, 523, 34, 49, 524, 34, 51, 525, 34, 53, 526, 34, 55, 527, 34, 57, 528, 34, 59, 529, 34, 61, 530  
 DATA 34, 63, 531, 34, 65, 532, 34, 67, 533, 34, 69, 534, 34, 71, 535, 34, 73, 536, 34, 75, 537, 34, 77, 538, 34, 79, 539, 34, 81, 540

DATA 34, 83, 541, 34, 85, 542, 34, 87, 543, 34, 89, 544, 34, 91, 545, 34, 93, 546, 34, 95, 547, 34, 97, 548, 34, 99, 549, 34, 101, 550  
DATA 34, 103, 551, 34, 105, 552, 34, 107, 553, 34, 109, 554, 34, 111, 555, 34, 113, 556, 34, 115, 557, 34, 117, 558, 34, 119, 559, 34, 121, 560  
DATA 34, 123, 561, 34, 125, 562, 34, 127, 563, 34, 129, 564, 34, 131, 565, 34, 133, 566, 34, 135, 567, 34, 137, 568, 34, 139, 569, 34, 141, 570  
DATA 34, 143, 571, 34, 145, 572, 34, 147, 573, 34, 149, 574, 34, 151, 575, 34, 153, 576, 34, 155, 577, 34, 157, 578, 34, 159, 579, 34, 161, 580  
DATA 34, 163, 581, 34, 165, 582, 35, 1, 583, 35, 3, 584, 35, 5, 585, 35, 7, 586, 35, 9, 587, 35, 11, 588, 35, 13, 589, 35, 15, 590  
DATA 35, 17, 591, 35, 19, 592, 35, 21, 593, 35, 23, 594, 35, 25, 595, 35, 27, 596, 35, 29, 597, 35, 31, 598, 35, 33, 599, 35, 35, 600  
DATA 35, 37, 601, 35, 39, 602, 35, 41, 603, 35, 43, 604, 35, 45, 605, 35, 47, 606, 35, 49, 607, 35, 51, 608, 35, 53, 609, 35, 55, 610  
DATA 35, 57, 611, 35, 59, 612, 35, 61, 613, 35, 63, 614, 35, 65, 615, 35, 67, 616, 35, 69, 617, 35, 71, 618, 35, 73, 619, 35, 75, 620  
DATA 35, 77, 621, 35, 78, 622, 35, 79, 623, 35, 81, 624, 35, 83, 625, 35, 85, 626, 35, 87, 627, 35, 89, 628, 35, 91, 629, 35, 93, 630  
DATA 35, 95, 631, 35, 97, 632, 35, 99, 633, 35, 101, 634, 35, 103, 635, 35, 105, 636, 35, 107, 637, 35, 109, 638, 35, 111, 639, 35, 113, 640  
DATA 35, 115, 641, 35, 117, 642, 35, 119, 643, 35, 121, 644, 35, 123, 645, 35, 125, 646, 35, 127, 647, 35, 129, 648, 35, 131, 649, 35, 133, 650  
DATA 35, 135, 651, 35, 137, 652, 35, 139, 653, 35, 141, 654, 41, 1, 655, 41, 3, 656, 41, 5, 657, 41, 7, 658, 41, 9, 659, 41, 11, 660  
DATA 41, 13, 661, 41, 15, 662, 41, 17, 663, 41, 19, 664, 41, 21, 665, 41, 23, 666, 41, 25, 667, 41, 27, 668, 41, 29, 669, 41, 31, 670  
DATA 41, 33, 671, 41, 35, 672, 41, 37, 673, 41, 39, 674, 41, 41, 675, 41, 43, 676, 41, 45, 677, 41, 47, 678, 41, 49, 679, 41, 51, 680  
DATA 41, 53, 681, 41, 55, 682, 41, 57, 683, 41, 59, 684, 41, 61, 685, 41, 63, 686, 41, 65, 687, 41, 67, 688, 41, 69, 689, 41, 71, 690  
DATA 41, 73, 691, 41, 75, 692, 41, 77, 693, 41, 79, 694, 41, 81, 695, 41, 83, 696, 41, 85, 697, 41, 87, 698, 41, 89, 699, 41, 91, 700  
DATA 41, 93, 701, 41, 95, 702, 41, 97, 703, 41, 99, 704, 41, 101, 705, 41, 103, 706, 41, 105, 707, 41, 107, 708, 41, 109, 709, 41, 111, 710  
DATA 41, 113, 711, 41, 115, 712, 41, 117, 713, 41, 119, 714, 41, 121, 715, 41, 123, 716, 41, 125, 717, 41, 127, 718, 41, 129, 719, 41, 131, 720  
DATA 41, 133, 721, 41, 135, 722, 41, 137, 723, 41, 139, 724, 41, 141, 725, 41, 143, 726, 41, 145, 727, 41, 147, 728, 41, 149, 729, 41, 151, 730  
DATA 41, 153, 731, 41, 155, 732, 41, 157, 733, 41, 159, 734, 41, 161, 735, 41, 163, 736, 41, 165, 737, 41, 167, 738, 41, 169, 739, 41, 171, 740  
DATA 41, 173, 741, 42, 1, 742, 42, 3, 743, 42, 5, 744, 42, 7, 745, 42, 9, 746, 42, 11, 747, 42, 13, 748, 42, 15, 749, 42, 17, 750  
DATA 42, 19, 751, 42, 21, 752, 42, 23, 753, 42, 25, 754, 42, 27, 755, 42, 29, 756, 42, 31, 757, 42, 33, 758, 42, 35, 759, 42, 37, 760  
DATA 42, 39, 761, 42, 41, 762, 42, 43, 763, 42, 45, 764, 42, 47, 765, 42, 49, 766, 42, 51, 767, 42, 53, 768, 42, 55, 769, 42, 57, 770  
DATA 42, 59, 771, 42, 61, 772, 42, 63, 773, 42, 65, 774, 42, 67, 775, 42, 69, 776, 42, 71, 777, 42, 73, 778, 42, 75, 779, 42, 77, 780  
DATA 42, 79, 781, 42, 81, 782, 42, 83, 783, 42, 85, 784, 42, 87, 785, 42, 89, 786, 42, 91, 787, 42, 93, 788, 42, 95, 789, 42, 97, 790  
DATA 42, 99, 791, 42, 101, 792, 42, 103, 793, 42, 105, 794, 42, 107, 795, 42, 109, 796, 42, 111, 797, 42, 113, 798, 42, 115, 799, 42, 117, 800  
DATA 42, 119, 801, 42, 121, 802, 42, 123, 803, 42, 125, 804, 42, 127, 805, 42, 129, 806, 42, 131, 807, 42, 133, 808, 42, 135, 809, 42, 137, 810  
DATA 42, 139, 811, 42, 141, 812, 42, 143, 813, 42, 145, 814, 42, 147, 815, 42, 149, 816, 42, 151, 817, 42, 153, 818, 42, 155, 819, 42, 157, 820  
DATA 42, 159, 821, 42, 161, 822, 42, 163, 823, 42, 165, 824, 42, 167, 825, 42, 169, 826, 42, 171, 827, 42, 173, 828, 42, 175, 829, 42, 177, 830  
DATA 42, 179, 831, 42, 181, 832, 42, 183, 833, 42, 185, 834, 42, 187, 835, 42, 189, 836, 42, 191, 837, 42, 193, 838, 42, 195, 839, 42, 197, 840  
DATA 43, 1, 841, 43, 3, 842, 43, 5, 843, 43, 7, 844, 43, 9, 845, 43, 11, 846, 43, 13, 847, 43, 15, 848, 43, 17, 849, 43, 19, 850

DATA 43, 21, 851, 43, 23, 852, 43, 25, 853, 43, 27, 854, 43, 29, 855, 43, 31, 856, 43, 33, 857, 43, 35, 858, 43, 37, 859, 43, 39, 860  
DATA 43, 41, 861, 43, 43, 862, 43, 45, 863, 43, 47, 864, 43, 49, 865, 43, 51, 866, 43, 53, 867, 43, 55, 868, 43, 57, 869, 43, 59, 870  
DATA 43, 61, 871, 43, 63, 872, 43, 65, 873, 43, 67, 874, 43, 69, 875, 43, 71, 876, 43, 73, 877, 43, 75, 878, 43, 77, 879, 43, 79, 880  
DATA 43, 81, 881, 43, 83, 882, 43, 85, 883, 43, 87, 884, 43, 89, 885, 43, 91, 886, 43, 93, 887, 43, 95, 888, 43, 97, 889, 43, 99, 890  
DATA 43, 101, 891, 43, 103, 892, 43, 105, 893, 43, 107, 894, 43, 109, 895, 43, 111, 896, 43, 113, 897, 43, 115, 898, 43, 117, 899, 43, 119, 900  
DATA 43, 121, 901, 43, 123, 902, 43, 125, 903, 43, 127, 904, 43, 129, 905, 43, 131, 906, 43, 133, 907, 43, 135, 908, 43, 137, 909, 43, 139, 910  
DATA 43, 141, 911, 43, 143, 912, 43, 145, 913, 43, 147, 914, 43, 149, 915, 43, 151, 916, 43, 153, 917, 43, 155, 918, 43, 157, 919, 43, 159, 920  
DATA 43, 161, 921, 43, 163, 922, 43, 165, 923, 43, 167, 924, 43, 169, 925, 43, 171, 926, 43, 173, 927, 43, 175, 928, 43, 177, 929, 43, 179, 930  
DATA 43, 181, 931, 43, 183, 932, 43, 185, 933, 43, 186, 934, 43, 187, 935, 43, 189, 936, 43, 195, 937, 43, 197, 938, 43, 199, 939, 43, 201, 940  
DATA 43, 203, 941, 43, 205, 942, 43, 207, 943, 43, 209, 944, 43, 211, 945, 43, 213, 946, 43, 215, 947, 43, 217, 948, 43, 219, 949, 43, 221, 950  
DATA 43, 223, 951, 43, 225, 952, 43, 227, 953, 43, 229, 954, 43, 510, 955, 44, 1, 956, 44, 3, 957, 44, 5, 958, 44, 7, 959, 44, 9, 960  
DATA 44, 11, 961, 44, 13, 962, 44, 15, 963, 44, 17, 964, 44, 19, 965, 44, 21, 966, 44, 23, 967, 44, 25, 968, 44, 27, 969, 44, 29, 970  
DATA 44, 31, 971, 44, 33, 972, 44, 35, 973, 44, 37, 974, 44, 39, 975, 44, 41, 976, 44, 43, 977, 44, 45, 978, 44, 47, 979, 44, 49, 980  
DATA 44, 51, 981, 44, 53, 982, 44, 55, 983, 44, 57, 984, 44, 59, 985, 44, 61, 986, 44, 63, 987, 44, 65, 988, 44, 67, 989, 44, 69, 990  
DATA 44, 71, 991, 44, 73, 992, 44, 75, 993, 44, 77, 994, 44, 79, 995, 44, 81, 996, 44, 83, 997, 44, 85, 998, 44, 87, 999, 44, 89, 1000  
DATA 44, 91, 1001, 44, 93, 1002, 44, 95, 1003, 44, 97, 1004, 44, 99, 1005, 44, 101, 1006, 44, 103, 1007, 44, 105, 1008, 45, 3, 1009, 45, 5, 1010  
DATA 45, 7, 1011, 45, 9, 1012, 45, 11, 1013, 45, 13, 1014, 45, 15, 1015, 45, 17, 1016, 45, 19, 1017, 45, 21, 1018, 45, 23, 1019, 45, 25, 1020  
DATA 45, 27, 1021, 45, 29, 1022, 45, 31, 1023, 45, 33, 1024, 45, 35, 1025, 45, 37, 1026, 45, 39, 1027, 45, 41, 1028, 45, 43, 1029, 45, 45, 1030  
DATA 45, 47, 1031, 45, 49, 1032, 45, 51, 1033, 45, 53, 1034, 45, 55, 1035, 45, 57, 1036, 45, 59, 1037, 45, 61, 1038, 45, 63, 1039, 45, 65, 1040  
DATA 45, 67, 1041, 45, 69, 1042, 45, 71, 1043, 45, 73, 1044, 45, 75, 1045, 45, 77, 1046, 45, 79, 1047, 45, 81, 1048, 45, 83, 1049, 45, 85, 1050  
DATA 45, 87, 1051, 45, 89, 1052, 45, 91, 1053, 45, 93, 1054, 45, 95, 1055, 45, 97, 1056, 45, 99, 1057, 45, 101, 1058, 45, 103, 1059, 45, 105, 1060  
DATA 45, 107, 1061, 45, 109, 1062, 45, 111, 1063, 45, 113, 1064, 45, 115, 1065, 45, 117, 1066, 45, 119, 1067, 45, 121, 1068, 45, 123, 1069, 45, 125, 1070  
DATA 45, 127, 1071, 45, 129, 1072, 45, 135, 1073, 45, 137, 1074, 46, 1, 1075, 46, 3, 1076, 46, 5, 1077, 46, 7, 1078, 46, 9, 1079, 46, 11, 1080  
DATA 46, 13, 1081, 46, 15, 1082, 46, 17, 1083, 46, 19, 1084, 46, 21, 1085, 46, 23, 1086, 46, 25, 1087, 46, 27, 1088, 46, 29, 1089, 46, 31, 1090  
DATA 46, 33, 1091, 46, 35, 1092, 46, 37, 1093, 46, 39, 1094, 46, 41, 1095, 46, 43, 1096, 46, 45, 1097, 46, 47, 1098, 46, 49, 1099, 46, 51, 1100  
DATA 46, 53, 1101, 46, 55, 1102, 46, 57, 1103, 46, 59, 1104, 46, 61, 1105, 46, 63, 1106, 46, 65, 1107, 46, 67, 1108, 46, 69, 1109, 46, 71, 1110  
DATA 46, 73, 1111, 46, 75, 1112, 46, 77, 1113, 46, 79, 1114, 46, 81, 1115, 46, 83, 1116, 46, 85, 1117, 46, 87, 1118, 46, 89, 1119, 46, 91, 1120  
DATA 46, 93, 1121, 46, 95, 1122, 46, 97, 1123, 46, 99, 1124, 46, 101, 1125, 46, 103, 1126, 46, 105, 1127, 46, 107, 1128, 46, 109, 1129, 46, 111, 1130  
DATA 46, 113, 1131, 46, 115, 1132, 46, 117, 1133, 46, 119, 1134, 46, 121, 1135, 46, 123, 1136, 46, 125, 1137, 46, 127, 1138, 46, 129, 1139, 46, 131, 1140  
DATA 46, 133, 1141, 46, 135, 1142, 46, 137, 1143, 46, 139, 1144, 46, 141, 1145, 46, 143, 1146, 46, 145, 1147, 46, 147, 1148, 46, 149, 1149, 46, 151, 1150  
DATA 46, 153, 1151, 46, 155, 1152, 46, 157, 1153, 46, 159, 1154, 46, 161, 1155, 46, 163, 1156, 46, 165, 1157, 46, 167, 1158, 46, 169, 1159, 46, 171, 1160

DATA 46, 173, 1161, 46, 175, 1162, 46, 177, 1163, 46, 179, 1164, 46, 181, 1165, 46, 183, 1166, 46, 185, 1167, 47, 1, 1168, 47, 3, 1169, 47, 5, 1170  
DATA 47, 7, 1171, 47, 9, 1172, 47, 11, 1173, 47, 13, 1174, 47, 15, 1175, 47, 17, 1176, 47, 19, 1177, 47, 21, 1178, 47, 23, 1179, 47, 25, 1180  
DATA 47, 27, 1181, 47, 29, 1182, 47, 31, 1183, 47, 33, 1184, 47, 35, 1185, 47, 37, 1186, 47, 39, 1187, 47, 41, 1188, 47, 43, 1189, 47, 45, 1190  
DATA 47, 47, 1191, 47, 49, 1192, 47, 51, 1193, 47, 53, 1194, 47, 55, 1195, 47, 57, 1196, 47, 59, 1197, 47, 61, 1198, 47, 63, 1199, 47, 65, 1200  
DATA 47, 67, 1201, 47, 69, 1202, 47, 71, 1203, 47, 73, 1204, 47, 75, 1205, 47, 77, 1206, 47, 79, 1207, 47, 81, 1208, 47, 83, 1209, 47, 85, 1210  
DATA 47, 87, 1211, 47, 89, 1212, 47, 91, 1213, 47, 93, 1214, 47, 95, 1215, 47, 97, 1216, 47, 99, 1217, 47, 101, 1218, 47, 103, 1219, 47, 105, 1220  
DATA 47, 107, 1221, 47, 109, 1222, 47, 111, 1223, 47, 113, 1224, 47, 115, 1225, 47, 117, 1226, 47, 119, 1227, 47, 121, 1228, 47, 123, 1229, 47, 125, 1230  
DATA 47, 127, 1231, 47, 129, 1232, 47, 131, 1233, 47, 133, 1234, 47, 135, 1235, 47, 137, 1236, 47, 139, 1237, 47, 141, 1238, 47, 143, 1239, 47, 145, 1240  
DATA 47, 147, 1241, 47, 149, 1242, 47, 151, 1243, 47, 153, 1244, 47, 155, 1245, 47, 157, 1246, 47, 159, 1247, 47, 161, 1248, 47, 163, 1249, 47, 165, 1250  
DATA 47, 167, 1251, 47, 169, 1252, 47, 171, 1253, 47, 173, 1254, 47, 175, 1255, 47, 177, 1256, 47, 179, 1257, 47, 181, 1258, 47, 183, 1259, 47, 185, 1260  
DATA 47, 187, 1261, 47, 189, 1262, 47, 191, 1263, 47, 193, 1264, 47, 195, 1265, 47, 197, 1266, 47, 199, 1267, 47, 201, 1268, 47, 203, 1269, 47, 205, 1270  
DATA 47, 207, 1271, 47, 209, 1272, 51, 1, 1273, 51, 3, 1274, 51, 5, 1275, 52, 1, 1276, 52, 3, 1277, 52, 5, 1278, 52, 9, 1279, 52, 11, 1280  
DATA 52, 13, 1281, 52, 15, 1282, 52, 17, 1283, 52, 19, 1284, 52, 21, 1285, 52, 23, 1286, 52, 25, 1287, 52, 27, 1288, 52, 29, 1289, 52, 31, 1290  
DATA 52, 33, 1291, 52, 35, 1292, 52, 37, 1293, 52, 39, 1294, 52, 41, 1295, 52, 43, 1296, 52, 45, 1297, 52, 47, 1298, 52, 50, 1299, 53, 1, 1300  
DATA 54, 1, 1301, 54, 3, 1302, 54, 5, 1303, 54, 7, 1304, 54, 9, 1305, 54, 11, 1306, 54, 13, 1307, 54, 15, 1308, 54, 17, 1309, 54, 19, 1310  
DATA 54, 21, 1311, 54, 23, 1312, 54, 25, 1313, 54, 27, 1314, 54, 29, 1315, 54, 31, 1316, 54, 33, 1317, 54, 35, 1318, 54, 36, 1319, 54, 37, 1320  
DATA 54, 41, 1321, 54, 43, 1322, 54, 45, 1323, 54, 47, 1324, 54, 49, 1325, 54, 51, 1326, 54, 53, 1327, 54, 57, 1328, 54, 59, 1329, 54, 61, 1330  
DATA 54, 63, 1331, 54, 65, 1332, 54, 67, 1333, 54, 69, 1334, 54, 71, 1335, 54, 73, 1336, 54, 75, 1337, 54, 77, 1338, 54, 79, 1339, 54, 81, 1340  
DATA 54, 83, 1341, 54, 85, 1342, 54, 87, 1343, 54, 89, 1344, 54, 91, 1345, 54, 93, 1346, 54, 95, 1347, 54, 97, 1348, 54, 99, 1349, 54, 101, 1350  
DATA 54, 103, 1351, 54, 105, 1352, 54, 107, 1353, 54, 109, 1354, 54, 111, 1355, 54, 113, 1356, 54, 115, 1357, 54, 117, 1358, 54, 119, 1359, 54, 121, 1360  
DATA 54, 125, 1361, 54, 127, 1362, 54, 131, 1363, 54, 133, 1364, 54, 135, 1365, 54, 137, 1366, 54, 139, 1367, 54, 141, 1368, 54, 143, 1369, 54, 145, 1370  
DATA 54, 147, 1371, 54, 149, 1372, 54, 153, 1373, 54, 155, 1374, 54, 157, 1375, 54, 159, 1376, 54, 161, 1377, 54, 163, 1378, 54, 165, 1379, 54, 167, 1380  
DATA 54, 169, 1381, 54, 171, 1382, 54, 173, 1383, 54, 175, 1384, 54, 177, 1385, 54, 179, 1386, 54, 181, 1387, 54, 183, 1388, 54, 185, 1389, 54, 187, 1390  
DATA 54, 191, 1391, 54, 193, 1392, 54, 195, 1393, 54, 197, 1394, 54, 199, 1395, 54, 510, 1396, 54, 515, 1397, 54, 520, 1398, 54, 530, 1399, 54, 540, 1400  
DATA 54, 550, 1401, 54, 560, 1402, 54, 570, 1403, 54, 580, 1404, 54, 590, 1405, 54, 595, 1406, 54, 600, 1407, 54, 610, 1408, 54, 620, 1409, 54, 630, 1410  
DATA 54, 640, 1411, 54, 650, 1412, 54, 660, 1413, 54, 670, 1414, 54, 678, 1415, 54, 680, 1416, 54, 683, 1417, 54, 685, 1418, 54, 690, 1419, 54, 700, 1420  
DATA 54, 710, 1421, 54, 720, 1422, 54, 730, 1423, 54, 735, 1424, 54, 740, 1425, 54, 750, 1426, 54, 760, 1427, 54, 770, 1428, 54, 775, 1429, 54, 780, 1430  
DATA 54, 790, 1431, 54, 800, 1432, 54, 810, 1433, 54, 820, 1434, 54, 830, 1435, 54, 840, 1436, 55, 1, 1437, 55, 3, 1438, 55, 5, 1439, 55, 7, 1440  
DATA 55, 9, 1441, 55, 11, 1442, 55, 13, 1443, 55, 15, 1444, 55, 17, 1445, 55, 19, 1446, 55, 21, 1447, 55, 23, 1448, 55, 25, 1449, 55, 27, 1450  
DATA 55, 29, 1451, 55, 31, 1452, 55, 33, 1453, 55, 35, 1454, 55, 37, 1455, 55, 39, 1456, 55, 41, 1457, 55, 43, 1458, 55, 45, 1459, 55, 47, 1460  
DATA 55, 49, 1461, 55, 51, 1462, 55, 53, 1463, 55, 55, 1464, 55, 57, 1465, 55, 59, 1466, 55, 61, 1467, 55, 63, 1468, 55, 65, 1469, 55, 67, 1470

DATA 55, 69, 1471, 55, 71, 1472, 55, 73, 1473, 55, 75, 1474, 55, 77, 1475, 55, 79, 1476, 55, 81, 1477, 55, 83, 1478, 55, 85, 1479, 55, 87, 1480  
DATA 55, 89, 1481, 55, 91, 1482, 55, 93, 1483, 55, 95, 1484, 55, 97, 1485, 55, 99, 1486, 55, 101, 1487, 55, 103, 1488, 55, 105, 1489, 55, 107, 1490  
DATA 55, 109, 1491, 56, 1, 1492, 56, 3, 1493, 56, 5, 1494, 56, 7, 1495, 56, 9, 1496, 56, 11, 1497, 56, 13, 1498, 56, 15, 1499, 56, 17, 1500  
DATA 56, 19, 1501, 56, 21, 1502, 56, 23, 1503, 56, 25, 1504, 56, 27, 1505, 56, 29, 1506, 56, 31, 1507, 56, 33, 1508, 56, 35, 1509, 56, 37, 1510  
DATA 56, 39, 1511, 56, 41, 1512, 56, 43, 1513, 56, 45, 1514, 56, 47, 1515, 56, 49, 1516, 56, 51, 1517, 56, 53, 1518, 56, 55, 1519, 56, 57, 1520  
DATA 56, 59, 1521, 56, 61, 1522, 56, 63, 1523, 56, 65, 1524, 56, 67, 1525, 56, 69, 1526, 56, 71, 1527, 56, 73, 1528, 56, 75, 1529, 56, 77, 1530  
DATA 56, 79, 1531, 56, 81, 1532, 56, 83, 1533, 56, 85, 1534, 56, 87, 1535, 56, 89, 1536, 56, 91, 1537, 56, 93, 1538, 56, 95, 1539, 56, 97, 1540  
DATA 56, 99, 1541, 56, 101, 1542, 56, 103, 1543, 56, 105, 1544, 56, 107, 1545, 56, 109, 1546, 56, 111, 1547, 56, 113, 1548, 56, 115, 1549, 56, 117, 1550  
DATA 56, 119, 1551, 56, 121, 1552, 56, 123, 1553, 56, 125, 1554, 56, 127, 1555, 56, 129, 1556, 56, 131, 1557, 56, 133, 1558, 56, 135, 1559, 56, 137, 1560  
DATA 56, 139, 1561, 56, 141, 1562, 56, 143, 1563, 56, 145, 1564, 56, 147, 1565, 56, 149, 1566, 56, 151, 1567, 56, 153, 1568, 56, 155, 1569, 56, 157, 1570  
DATA 56, 159, 1571, 56, 161, 1572, 56, 163, 1573, 56, 165, 1574, 56, 167, 1575, 56, 169, 1576, 56, 171, 1577, 56, 173, 1578, 56, 175, 1579, 56, 177, 1580  
DATA 56, 179, 1581, 56, 181, 1582, 56, 183, 1583, 56, 185, 1584, 56, 187, 1585, 56, 189, 1586, 56, 191, 1587, 56, 193, 1588, 56, 195, 1589, 56, 197, 1590  
DATA 56, 199, 1591, 57, 1, 1592, 57, 3, 1593, 57, 5, 1594, 57, 7, 1595, 57, 9, 1596, 57, 11, 1597, 57, 13, 1598, 57, 15, 1599, 57, 17, 1600  
DATA 57, 19, 1601, 57, 21, 1602, 57, 23, 1603, 57, 25, 1604, 57, 27, 1605, 57, 29, 1606, 57, 31, 1607, 57, 33, 1608, 57, 35, 1609, 57, 37, 1610  
DATA 57, 39, 1611, 57, 41, 1612, 57, 43, 1613, 57, 45, 1614, 57, 47, 1615, 57, 49, 1616, 57, 51, 1617, 57, 53, 1618, 57, 55, 1619, 57, 57, 1620  
DATA 57, 59, 1621, 57, 61, 1622, 57, 63, 1623, 57, 65, 1624, 57, 67, 1625, 57, 69, 1626, 57, 71, 1627, 57, 73, 1628, 57, 75, 1629, 57, 77, 1630  
DATA 57, 79, 1631, 57, 81, 1632, 57, 83, 1633, 57, 85, 1634, 57, 87, 1635, 57, 89, 1636, 57, 91, 1637, 58, 1, 1638, 58, 3, 1639, 58, 5, 1640  
DATA 58, 7, 1641, 58, 9, 1642, 58, 11, 1643, 58, 13, 1644, 58, 15, 1645, 58, 17, 1646, 58, 19, 1647, 58, 21, 1648, 58, 23, 1649, 58, 25, 1650  
DATA 58, 27, 1651, 58, 29, 1652, 58, 31, 1653, 58, 33, 1654, 58, 35, 1655, 58, 37, 1656, 58, 39, 1657, 58, 43, 1658, 58, 45, 1659, 58, 47, 1660  
DATA 58, 49, 1661, 58, 51, 1662, 58, 53, 1663, 58, 55, 1664, 58, 57, 1665, 58, 59, 1666, 58, 61, 1667, 58, 63, 1668, 58, 65, 1669, 58, 67, 1670  
DATA 58, 69, 1671, 58, 71, 1672, 58, 73, 1673, 58, 75, 1674, 58, 77, 1675, 58, 79, 1676, 58, 81, 1677, 58, 83, 1678, 58, 85, 1679, 58, 87, 1680  
DATA 58, 89, 1681, 58, 91, 1682, 58, 93, 1683, 58, 95, 1684, 58, 97, 1685, 58, 99, 1686, 58, 101, 1687, 58, 103, 1688, 58, 105, 1689, 58, 107, 1690  
DATA 58, 109, 1691, 58, 111, 1692, 58, 113, 1693, 58, 115, 1694, 58, 117, 1695, 58, 119, 1696, 58, 121, 1697, 58, 123, 1698, 58, 125, 1699, 58, 127, 1700  
DATA 58, 129, 1701, 58, 131, 1702, 58, 133, 1703, 58, 135, 1704, 58, 137, 1705, 58, 139, 1706, 58, 141, 1707, 58, 143, 1708, 58, 145, 1709, 58, 147, 1710  
DATA 58, 149, 1711, 58, 151, 1712, 58, 153, 1713, 58, 155, 1714, 58, 157, 1715, 58, 159, 1716, 58, 161, 1717, 58, 163, 1718, 58, 165, 1719, 58, 167, 1720  
DATA 58, 169, 1721, 58, 171, 1722, 58, 173, 1723, 58, 175, 1724, 58, 177, 1725, 58, 179, 1726, 58, 181, 1727, 58, 183, 1728, 58, 185, 1729, 58, 187, 1730  
DATA 58, 189, 1731, 58, 191, 1732, 58, 193, 1733, 58, 195, 1734, 58, 197, 1735, 58, 199, 1736, 58, 201, 1737, 58, 205, 1738, 58, 207, 1739, 58, 209, 1740  
DATA 58, 211, 1741, 58, 213, 1742, 58, 215, 1743, 58, 217, 1744, 58, 219, 1745, 58, 221, 1746, 58, 223, 1747, 58, 225, 1748, 58, 227, 1749, 58, 229, 1750  
DATA 58, 231, 1751, 58, 233, 1752, 58, 235, 1753, 58, 237, 1754, 58, 239, 1755, 58, 241, 1756, 58, 243, 1757, 58, 245, 1758, 58, 247, 1759, 58, 249, 1760  
DATA 58, 251, 1761, 58, 253, 1762, 58, 255, 1763, 58, 257, 1764, 58, 259, 1765, 58, 261, 1766, 58, 263, 1767, 58, 265, 1768, 58, 267, 1769, 58, 269, 1770  
DATA 58, 271, 1771, 58, 273, 1772, 58, 275, 1773, 58, 277, 1774, 58, 279, 1775, 58, 281, 1776, 58, 283, 1777, 58, 285, 1778, 58, 287, 1779, 58, 289, 1780

DATA 58, 291, 1781, 58, 293, 1782, 58, 295, 1783, 58, 297, 1784, 58, 299, 1785, 58, 301, 1786, 58, 303, 1787, 58, 305, 1788, 58, 307, 1789, 58, 309, 1790  
DATA 58, 311, 1791, 58, 313, 1792, 58, 315, 1793, 58, 317, 1794, 58, 319, 1795, 58, 321, 1796, 59, 1, 1797, 59, 3, 1798, 59, 5, 1799, 59, 7, 1800  
DATA 59, 9, 1801, 59, 11, 1802, 59, 13, 1803, 59, 15, 1804, 59, 17, 1805, 59, 19, 1806, 59, 21, 1807, 59, 23, 1808, 59, 25, 1809, 59, 27, 1810  
DATA 59, 29, 1811, 59, 31, 1812, 59, 33, 1813, 59, 35, 1814, 59, 37, 1815, 59, 39, 1816, 59, 41, 1817, 59, 43, 1818, 59, 45, 1819, 59, 47, 1820  
DATA 59, 49, 1821, 59, 51, 1822, 59, 53, 1823, 59, 55, 1824, 59, 57, 1825, 59, 59, 1826, 59, 61, 1827, 59, 63, 1828, 59, 65, 1829, 59, 67, 1830  
DATA 59, 69, 1831, 59, 71, 1832, 59, 73, 1833, 59, 75, 1834, 59, 77, 1835, 59, 79, 1836, 59, 81, 1837, 59, 83, 1838, 59, 85, 1839, 59, 87, 1840  
DATA 59, 89, 1841, 59, 91, 1842, 59, 93, 1843, 59, 95, 1844, 59, 97, 1845, 59, 99, 1846, 59, 101, 1847, 59, 103, 1848, 59, 105, 1849, 59, 107, 1850  
DATA 59, 109, 1851, 59, 111, 1852, 59, 113, 1853, 59, 115, 1854, 59, 117, 1855, 59, 119, 1856, 59, 121, 1857, 59, 123, 1858, 59, 125, 1859, 59, 127, 1860  
DATA 59, 129, 1861, 59, 131, 1862, 59, 133, 1863, 61, 1, 1864, 61, 3, 1865, 61, 5, 1866, 61, 7, 1867, 61, 9, 1868, 61, 11, 1869, 61, 13, 1870  
DATA 61, 15, 1871, 61, 17, 1872, 61, 19, 1873, 61, 21, 1874, 61, 23, 1875, 61, 25, 1876, 61, 27, 1877, 61, 29, 1878, 61, 31, 1879, 61, 33, 1880  
DATA 61, 35, 1881, 61, 37, 1882, 61, 39, 1883, 61, 41, 1884, 61, 43, 1885, 61, 45, 1886, 61, 47, 1887, 61, 49, 1888, 61, 51, 1889, 61, 53, 1890  
DATA 61, 55, 1891, 61, 57, 1892, 61, 59, 1893, 61, 61, 1894, 61, 63, 1895, 61, 65, 1896, 61, 67, 1897, 61, 69, 1898, 61, 71, 1899, 61, 73, 1900  
DATA 61, 75, 1901, 61, 77, 1902, 61, 79, 1903, 61, 81, 1904, 61, 83, 1905, 61, 85, 1906, 61, 87, 1907, 61, 89, 1908, 61, 91, 1909, 61, 93, 1910  
DATA 61, 95, 1911, 61, 97, 1912, 61, 99, 1913, 61, 101, 1914, 61, 103, 1915, 61, 105, 1916, 61, 107, 1917, 61, 109, 1918, 61, 111, 1919, 61, 113, 1920  
DATA 61, 115, 1921, 61, 117, 1922, 61, 119, 1923, 61, 121, 1924, 61, 123, 1925, 61, 125, 1926, 61, 127, 1927, 61, 129, 1928, 61, 131, 1929, 61, 133, 1930  
DATA 61, 135, 1931, 61, 137, 1932, 61, 139, 1933, 61, 141, 1934, 61, 143, 1935, 61, 145, 1936, 61, 147, 1937, 61, 149, 1938, 61, 151, 1939, 61, 153, 1940  
DATA 61, 155, 1941, 61, 157, 1942, 61, 159, 1943, 61, 161, 1944, 61, 163, 1945, 61, 165, 1946, 61, 167, 1947, 61, 169, 1948, 61, 171, 1949, 61, 173, 1950  
DATA 61, 175, 1951, 61, 177, 1952, 61, 179, 1953, 61, 181, 1954, 61, 183, 1955, 61, 185, 1956, 61, 187, 1957, 61, 189, 1958, 61, 191, 1959, 61, 193, 1960  
DATA 61, 195, 1961, 61, 197, 1962, 61, 199, 1963, 61, 201, 1964, 61, 203, 1965, 61, 205, 1966, 61, 207, 1967, 61, 209, 1968, 61, 211, 1969, 61, 213, 1970  
DATA 61, 215, 1971, 61, 217, 1972, 61, 219, 1973, 61, 221, 1974, 61, 223, 1975, 61, 225, 1976, 61, 227, 1977, 61, 229, 1978, 61, 231, 1979, 61, 233, 1980  
DATA 61, 235, 1981, 61, 237, 1982, 61, 239, 1983, 62, 1, 1984, 62, 3, 1985, 62, 5, 1986, 62, 7, 1987, 62, 9, 1988, 62, 11, 1989, 62, 13, 1990  
DATA 62, 15, 1991, 62, 17, 1992, 62, 19, 1993, 62, 21, 1994, 62, 23, 1995, 62, 25, 1996, 62, 27, 1997, 62, 29, 1998, 62, 31, 1999, 62, 33, 2000  
DATA 62, 35, 2001, 62, 37, 2002, 62, 39, 2003, 62, 41, 2004, 62, 43, 2005, 62, 45, 2006, 62, 47, 2007, 62, 49, 2008, 62, 51, 2009, 62, 53, 2010  
DATA 62, 55, 2011, 62, 57, 2012, 62, 59, 2013, 62, 61, 2014, 62, 63, 2015, 62, 65, 2016, 62, 67, 2017, 62, 69, 2018, 62, 71, 2019, 62, 73, 2020  
DATA 62, 75, 2021, 62, 77, 2022, 62, 79, 2023, 62, 81, 2024, 62, 83, 2025, 62, 85, 2026, 62, 87, 2027, 62, 89, 2028, 62, 91, 2029, 62, 93, 2030  
DATA 62, 95, 2031, 62, 97, 2032, 62, 99, 2033, 62, 101, 2034, 62, 103, 2035, 62, 105, 2036, 62, 107, 2037, 62, 109, 2038, 62, 111, 2039, 62, 113, 2040  
DATA 62, 115, 2041, 62, 117, 2042, 62, 119, 2043, 62, 121, 2044, 62, 123, 2045, 62, 125, 2046, 62, 127, 2047, 62, 129, 2048, 62, 131, 2049, 62, 133, 2050  
DATA 62, 135, 2051, 62, 137, 2052, 62, 139, 2053, 62, 141, 2054, 62, 143, 2055, 62, 145, 2056, 62, 147, 2057, 62, 149, 2058, 62, 151, 2059, 62, 153, 2060  
DATA 62, 155, 2061, 62, 157, 2062, 62, 159, 2063, 62, 161, 2064, 62, 163, 2065, 62, 165, 2066, 62, 167, 2067, 62, 169, 2068, 62, 171, 2069, 62, 173, 2070  
DATA 62, 175, 2071, 62, 177, 2072, 62, 179, 2073, 62, 181, 2074, 62, 183, 2075, 62, 185, 2076, 62, 187, 2077, 62, 189, 2078, 63, 1, 2079, 63, 3, 2080  
DATA 63, 5, 2081, 63, 7, 2082, 63, 9, 2083, 63, 11, 2084, 63, 13, 2085, 63, 15, 2086, 63, 17, 2087, 63, 19, 2088, 63, 21, 2089, 63, 23, 2090

DATA 63, 25, 2091, 63, 27, 2092, 63, 29, 2093, 63, 31, 2094, 63, 33, 2095, 63, 35, 2096, 63, 37, 2097, 63, 39, 2098, 63, 41, 2099, 63, 43, 2100  
DATA 63, 45, 2101, 63, 47, 2102, 63, 49, 2103, 63, 51, 2104, 63, 53, 2105, 63, 55, 2106, 63, 57, 2107, 63, 59, 2108, 63, 61, 2109, 63, 63, 2110  
DATA 63, 65, 2111, 63, 67, 2112, 63, 69, 2113, 63, 71, 2114, 63, 73, 2115, 63, 75, 2116, 63, 77, 2117, 63, 79, 2118, 63, 81, 2119, 63, 83, 2120  
DATA 63, 85, 2121, 63, 87, 2122, 63, 89, 2123, 63, 91, 2124, 63, 93, 2125, 63, 95, 2126, 63, 97, 2127, 63, 99, 2128, 63, 101, 2129, 63, 103, 2130  
DATA 63, 105, 2131, 63, 107, 2132, 63, 109, 2133, 63, 111, 2134, 63, 113, 2135, 63, 115, 2136, 63, 117, 2137, 63, 119, 2138, 63, 121, 2139, 63, 123, 2140  
DATA 63, 125, 2141, 63, 127, 2142, 63, 129, 2143, 63, 131, 2144, 63, 133, 2145, 64, 1, 2146, 64, 3, 2147, 64, 5, 2148, 64, 7, 2149, 64, 9, 2150  
DATA 64, 11, 2151, 64, 13, 2152, 64, 15, 2153, 64, 17, 2154, 64, 19, 2155, 64, 21, 2156, 64, 23, 2157, 64, 25, 2158, 64, 27, 2159, 64, 29, 2160  
DATA 64, 31, 2161, 64, 33, 2162, 64, 35, 2163, 64, 37, 2164, 64, 39, 2165, 64, 41, 2166, 64, 43, 2167, 64, 45, 2168, 64, 47, 2169, 64, 49, 2170  
DATA 64, 51, 2171, 64, 53, 2172, 64, 55, 2173, 64, 57, 2174, 64, 59, 2175, 64, 61, 2176, 64, 63, 2177, 64, 65, 2178, 64, 67, 2179, 64, 69, 2180  
DATA 64, 71, 2181, 64, 73, 2182, 64, 75, 2183, 64, 77, 2184, 64, 79, 2185, 64, 81, 2186, 64, 83, 2187, 64, 85, 2188, 64, 87, 2189, 64, 89, 2190  
DATA 64, 91, 2191, 64, 93, 2192, 64, 95, 2193, 64, 97, 2194, 64, 99, 2195, 64, 101, 2196, 64, 103, 2197, 64, 105, 2198, 64, 107, 2199, 64, 109, 2200  
DATA 64, 111, 2201, 64, 113, 2202, 64, 115, 2203, 64, 117, 2204, 64, 119, 2205, 64, 121, 2206, 64, 123, 2207, 64, 125, 2208, 64, 127, 2209, 64, 129, 2210  
DATA 64, 131, 2211, 64, 133, 2212, 64, 135, 2213, 64, 137, 2214, 64, 139, 2215, 64, 141, 2216, 64, 143, 2217, 64, 145, 2218, 64, 147, 2219, 64, 149, 2220  
DATA 64, 151, 2221, 64, 153, 2222, 64, 155, 2223, 64, 157, 2224, 64, 159, 2225, 64, 161, 2226, 64, 163, 2227, 71, 1, 2228, 71, 3, 2229, 71, 5, 2230  
DATA 71, 7, 2231, 71, 9, 2232, 71, 11, 2233, 71, 13, 2234, 71, 15, 2235, 71, 17, 2236, 71, 19, 2237, 71, 21, 2238, 71, 23, 2239, 71, 25, 2240  
DATA 71, 27, 2241, 71, 29, 2242, 71, 31, 2243, 71, 33, 2244, 71, 35, 2245, 71, 37, 2246, 71, 39, 2247, 71, 41, 2248, 71, 43, 2249, 71, 45, 2250  
DATA 71, 47, 2251, 71, 49, 2252, 71, 51, 2253, 71, 53, 2254, 71, 55, 2255, 71, 57, 2256, 71, 59, 2257, 71, 61, 2258, 71, 63, 2259, 71, 65, 2260  
DATA 71, 67, 2261, 71, 69, 2262, 71, 71, 2263, 71, 73, 2264, 71, 75, 2265, 71, 77, 2266, 71, 79, 2267, 71, 81, 2268, 71, 83, 2269, 71, 85, 2270  
DATA 71, 87, 2271, 71, 89, 2272, 71, 91, 2273, 71, 93, 2274, 71, 95, 2275, 71, 97, 2276, 71, 99, 2277, 71, 101, 2278, 71, 103, 2279, 71, 105, 2280  
DATA 71, 107, 2281, 71, 109, 2282, 71, 111, 2283, 71, 113, 2284, 71, 115, 2285, 71, 117, 2286, 71, 119, 2287, 71, 121, 2288, 71, 123, 2289, 71, 125, 2290  
DATA 71, 127, 2291, 71, 129, 2292, 71, 131, 2293, 71, 133, 2294, 71, 135, 2295, 71, 137, 2296, 71, 139, 2297, 71, 141, 2298, 71, 143, 2299, 71, 145, 2300  
DATA 71, 147, 2301, 71, 149, 2302, 72, 1, 2303, 72, 3, 2304, 72, 5, 2305, 72, 7, 2306, 72, 9, 2307, 72, 11, 2308, 72, 13, 2309, 72, 15, 2310  
DATA 72, 17, 2311, 72, 19, 2312, 72, 21, 2313, 72, 23, 2314, 72, 25, 2315, 72, 27, 2316, 72, 29, 2317, 72, 31, 2318, 72, 33, 2319, 72, 35, 2320  
DATA 72, 37, 2321, 72, 39, 2322, 72, 41, 2323, 72, 43, 2324, 72, 45, 2325, 72, 47, 2326, 72, 49, 2327, 72, 51, 2328, 72, 53, 2329, 72, 55, 2330  
DATA 72, 57, 2331, 72, 59, 2332, 72, 61, 2333, 72, 63, 2334, 72, 65, 2335, 72, 67, 2336, 72, 69, 2337, 72, 71, 2338, 72, 73, 2339, 72, 75, 2340  
DATA 72, 77, 2341, 72, 79, 2342, 72, 81, 2343, 72, 83, 2344, 72, 85, 2345, 72, 87, 2346, 72, 89, 2347, 72, 91, 2348, 72, 93, 2349, 72, 95, 2350  
DATA 72, 97, 2351, 72, 99, 2352, 72, 101, 2353, 72, 103, 2354, 72, 105, 2355, 72, 107, 2356, 72, 109, 2357, 72, 111, 2358, 72, 113, 2359, 72, 115, 2360  
DATA 72, 117, 2361, 72, 119, 2362, 72, 121, 2363, 72, 123, 2364, 72, 125, 2365, 72, 127, 2366, 73, 1, 2367, 73, 3, 2368, 73, 5, 2369, 73, 7, 2370  
DATA 73, 9, 2371, 73, 11, 2372, 73, 13, 2373, 73, 15, 2374, 73, 17, 2375, 73, 19, 2376, 73, 21, 2377, 73, 23, 2378, 73, 25, 2379, 73, 27, 2380  
DATA 73, 29, 2381, 73, 31, 2382, 73, 33, 2383, 73, 35, 2384, 73, 37, 2385, 73, 39, 2386, 73, 41, 2387, 73, 43, 2388, 73, 45, 2389, 73, 47, 2390  
DATA 73, 49, 2391, 73, 51, 2392, 73, 53, 2393, 73, 55, 2394, 73, 57, 2395, 73, 59, 2396, 73, 61, 2397, 73, 63, 2398, 73, 65, 2399, 73, 67, 2400

DATA 73, 69, 2401, 73, 71, 2402, 73, 73, 2403, 73, 75, 2404, 73, 77, 2405, 73, 79, 2406, 73, 81, 2407, 73, 83, 2408, 73, 85, 2409, 73, 87, 2410  
DATA 73, 89, 2411, 73, 91, 2412, 73, 93, 2413, 73, 95, 2414, 73, 97, 2415, 73, 99, 2416, 73, 101, 2417, 73, 103, 2418, 73, 105, 2419, 73, 107, 2420  
DATA 73, 109, 2421, 73, 111, 2422, 73, 113, 2423, 73, 115, 2424, 73, 117, 2425, 73, 119, 2426, 73, 121, 2427, 73, 123, 2428, 73, 125, 2429, 73, 127, 2430  
DATA 73, 129, 2431, 73, 131, 2432, 73, 133, 2433, 73, 135, 2434, 73, 137, 2435, 73, 139, 2436, 73, 141, 2437, 73, 143, 2438, 73, 145, 2439, 73, 147, 2440  
DATA 73, 149, 2441, 73, 151, 2442, 73, 153, 2443, 74, 1, 2444, 74, 3, 2445, 74, 5, 2446, 74, 7, 2447, 74, 9, 2448, 74, 11, 2449, 74, 13, 2450  
DATA 74, 15, 2451, 74, 17, 2452, 74, 19, 2453, 74, 21, 2454, 74, 23, 2455, 74, 25, 2456, 74, 27, 2457, 74, 29, 2458, 74, 31, 2459, 74, 33, 2460  
DATA 74, 35, 2461, 74, 37, 2462, 74, 39, 2463, 74, 41, 2464, 74, 43, 2465, 74, 45, 2466, 74, 47, 2467, 74, 49, 2468, 74, 51, 2469, 74, 53, 2470  
DATA 74, 55, 2471, 74, 57, 2472, 74, 59, 2473, 74, 61, 2474, 74, 63, 2475, 74, 65, 2476, 74, 67, 2477, 74, 69, 2478, 74, 71, 2479, 74, 73, 2480  
DATA 74, 75, 2481, 74, 77, 2482, 74, 79, 2483, 74, 81, 2484, 74, 83, 2485, 74, 85, 2486, 74, 87, 2487, 74, 89, 2488, 74, 91, 2489, 74, 93, 2490  
DATA 74, 95, 2491, 74, 97, 2492, 74, 99, 2493, 74, 101, 2494, 74, 103, 2495, 74, 105, 2496, 74, 107, 2497, 74, 109, 2498, 74, 111, 2499, 74, 113, 2500  
DATA 74, 115, 2501, 74, 117, 2502, 74, 119, 2503, 74, 121, 2504, 74, 123, 2505, 74, 125, 2506, 74, 127, 2507, 74, 129, 2508, 74, 131, 2509, 74, 133, 2510  
DATA 74, 135, 2511, 74, 137, 2512, 74, 139, 2513, 74, 141, 2514, 74, 143, 2515, 74, 145, 2516, 74, 147, 2517, 74, 149, 2518, 74, 151, 2519, 74, 153, 2520  
DATA 74, 155, 2521, 74, 157, 2522, 74, 159, 2523, 74, 161, 2524, 74, 163, 2525, 74, 165, 2526, 74, 167, 2527, 74, 169, 2528, 74, 171, 2529, 74, 173, 2530  
DATA 74, 175, 2531, 74, 177, 2532, 74, 179, 2533, 74, 181, 2534, 74, 183, 2535, 74, 185, 2536, 74, 187, 2537, 74, 189, 2538, 74, 191, 2539, 74, 193, 2540  
DATA 74, 195, 2541, 74, 197, 2542, 74, 199, 2543, 74, 201, 2544, 74, 203, 2545, 74, 205, 2546, 74, 207, 2547, 74, 209, 2548, 74, 211, 2549, 74, 213, 2550  
DATA 74, 215, 2551, 74, 217, 2552, 74, 219, 2553, 74, 221, 2554, 74, 223, 2555, 74, 225, 2556, 74, 227, 2557, 74, 229, 2558, 74, 231, 2559, 74, 233, 2560  
DATA 74, 235, 2561, 74, 237, 2562, 74, 239, 2563, 74, 241, 2564, 74, 243, 2565, 74, 245, 2566, 74, 247, 2567, 74, 249, 2568, 74, 251, 2569, 74, 253, 2570  
DATA 74, 255, 2571, 74, 257, 2572, 74, 259, 2573, 74, 261, 2574, 74, 263, 2575, 74, 265, 2576, 74, 267, 2577, 74, 269, 2578, 74, 271, 2579, 74, 273, 2580  
DATA 74, 275, 2581, 74, 277, 2582, 74, 279, 2583, 74, 281, 2584, 74, 283, 2585, 74, 285, 2586, 74, 287, 2587, 74, 289, 2588, 74, 291, 2589, 74, 293, 2590  
DATA 74, 295, 2591, 74, 297, 2592, 74, 299, 2593, 74, 301, 2594, 74, 303, 2595, 74, 305, 2596, 74, 307, 2597, 74, 309, 2598, 74, 311, 2599, 74, 313, 2600  
DATA 74, 315, 2601, 74, 317, 2602, 74, 319, 2603, 74, 321, 2604, 74, 323, 2605, 74, 325, 2606, 74, 327, 2607, 74, 329, 2608, 74, 331, 2609, 74, 333, 2610  
DATA 74, 335, 2611, 74, 337, 2612, 74, 339, 2613, 74, 341, 2614, 74, 343, 2615, 74, 345, 2616, 74, 347, 2617, 74, 349, 2618, 74, 351, 2619, 74, 353, 2620  
DATA 74, 355, 2621, 74, 357, 2622, 74, 359, 2623, 74, 361, 2624, 74, 363, 2625, 74, 365, 2626, 74, 367, 2627, 74, 369, 2628, 74, 371, 2629, 74, 373, 2630  
DATA 74, 375, 2631, 74, 377, 2632, 74, 379, 2633, 74, 381, 2634, 74, 383, 2635, 74, 385, 2636, 74, 387, 2637, 74, 389, 2638, 74, 391, 2639, 74, 393, 2640  
DATA 74, 395, 2641, 74, 397, 2642, 74, 399, 2643, 74, 401, 2644, 74, 403, 2645, 74, 405, 2646, 74, 407, 2647, 74, 409, 2648, 74, 411, 2649, 74, 413, 2650  
DATA 74, 415, 2651, 74, 417, 2652, 74, 419, 2653, 74, 421, 2654, 74, 423, 2655, 74, 425, 2656, 74, 427, 2657, 74, 429, 2658, 74, 431, 2659, 74, 433, 2660  
DATA 74, 435, 2661, 74, 437, 2662, 74, 439, 2663, 74, 441, 2664, 74, 443, 2665, 74, 445, 2666, 74, 447, 2667, 74, 449, 2668, 74, 451, 2669, 74, 453, 2670  
DATA 74, 455, 2671, 74, 457, 2672, 74, 459, 2673, 74, 461, 2674, 74, 463, 2675, 74, 465, 2676, 74, 467, 2677, 74, 469, 2678, 74, 471, 2679, 74, 473, 2680  
DATA 74, 475, 2681, 74, 477, 2682, 74, 479, 2683, 74, 481, 2684, 74, 483, 2685, 74, 485, 2686, 74, 487, 2687, 74, 489, 2688, 74, 491, 2689, 74, 493, 2690  
DATA 74, 495, 2691, 74, 497, 2692, 74, 499, 2693, 74, 501, 2694, 74, 503, 2695, 74, 505, 2696, 74, 507, 2697, 81, 1, 2698, 81, 3, 2699, 81, 5, 2700  
DATA 81, 7, 2701, 81, 9, 2702, 81, 11, 2703, 81, 13, 2704, 81, 15, 2705, 81, 17, 2706, 81, 19, 2707, 81, 21, 2708, 81, 23, 2709, 81, 25, 2710

DATA 81, 27, 2711, 81, 29, 2712, 81, 31, 2713, 81, 33, 2714, 81, 35, 2715, 81, 37, 2716, 81, 39, 2717, 81, 41, 2718, 81, 43, 2719, 81, 45, 2720  
DATA 81, 47, 2721, 81, 49, 2722, 81, 51, 2723, 81, 53, 2724, 81, 55, 2725, 81, 57, 2726, 81, 59, 2727, 81, 61, 2728, 81, 63, 2729, 81, 65, 2730  
DATA 81, 67, 2731, 81, 69, 2732, 81, 71, 2733, 81, 73, 2734, 81, 75, 2735, 81, 77, 2736, 81, 79, 2737, 81, 81, 2738, 81, 83, 2739, 81, 85, 2740  
DATA 81, 87, 2741, 81, 89, 2742, 81, 91, 2743, 81, 93, 2744, 81, 95, 2745, 81, 97, 2746, 81, 99, 2747, 81, 101, 2748, 81, 103, 2749, 81, 105, 2750  
DATA 81, 107, 2751, 81, 109, 2752, 81, 111, 2753, 81, 113, 2754, 82, 1, 2755, 82, 3, 2756, 82, 5, 2757, 82, 7, 2758, 82, 9, 2759, 82, 11, 2760  
DATA 82, 13, 2761, 82, 15, 2762, 82, 17, 2763, 82, 19, 2764, 82, 21, 2765, 82, 23, 2766, 82, 25, 2767, 82, 27, 2768, 82, 29, 2769, 82, 31, 2770  
DATA 82, 33, 2771, 82, 35, 2772, 82, 37, 2773, 82, 39, 2774, 82, 41, 2775, 82, 43, 2776, 82, 45, 2777, 82, 47, 2778, 82, 49, 2779, 82, 51, 2780  
DATA 82, 53, 2781, 82, 55, 2782, 82, 57, 2783, 82, 59, 2784, 82, 61, 2785, 82, 63, 2786, 82, 65, 2787, 82, 67, 2788, 82, 69, 2789, 82, 71, 2790  
DATA 82, 73, 2791, 82, 75, 2792, 82, 77, 2793, 82, 79, 2794, 82, 81, 2795, 82, 83, 2796, 82, 85, 2797, 82, 87, 2798, 83, 1, 2799, 83, 3, 2800  
DATA 83, 5, 2801, 83, 7, 2802, 83, 9, 2803, 83, 11, 2804, 83, 13, 2805, 83, 15, 2806, 83, 17, 2807, 83, 19, 2808, 83, 21, 2809, 83, 23, 2810  
DATA 83, 25, 2811, 83, 27, 2812, 83, 29, 2813, 83, 31, 2814, 83, 33, 2815, 83, 35, 2816, 83, 37, 2817, 83, 39, 2818, 83, 41, 2819, 83, 43, 2820  
DATA 83, 45, 2821, 84, 1, 2822, 84, 3, 2823, 84, 5, 2824, 84, 7, 2825, 84, 9, 2826, 84, 11, 2827, 84, 13, 2828, 84, 15, 2829, 84, 17, 2830  
DATA 84, 19, 2831, 84, 21, 2832, 84, 23, 2833, 84, 25, 2834, 84, 27, 2835, 84, 29, 2836, 84, 31, 2837, 84, 33, 2838, 84, 35, 2839, 84, 37, 2840  
DATA 84, 39, 2841, 84, 41, 2842, 84, 43, 2843, 84, 45, 2844, 84, 47, 2845, 84, 49, 2846, 84, 51, 2847, 84, 53, 2848, 84, 55, 2849, 84, 57, 2850  
DATA 84, 59, 2851, 84, 61, 2852, 84, 63, 2853, 84, 65, 2854, 84, 67, 2855, 84, 69, 2856, 84, 71, 2857, 84, 73, 2858, 84, 75, 2859, 84, 77, 2860  
DATA 84, 79, 2861, 84, 81, 2862, 84, 83, 2863, 84, 85, 2864, 84, 87, 2865, 84, 89, 2866, 84, 91, 2867, 84, 93, 2868, 84, 95, 2869, 84, 97, 2870  
DATA 84, 99, 2871, 84, 101, 2872, 84, 103, 2873, 84, 105, 2874, 84, 107, 2875, 84, 109, 2876, 84, 111, 2877, 84, 113, 2878, 84, 115, 2879, 84, 117, 2880  
DATA 84, 119, 2881, 84, 121, 2882, 84, 123, 2883, 84, 125, 2884, 85, 1, 2885, 85, 3, 2886, 85, 5, 2887, 85, 6, 2888, 85, 7, 2889, 85, 9, 2890  
DATA 85, 11, 2891, 85, 13, 2892, 85, 15, 2893, 85, 17, 2894, 85, 19, 2895, 85, 21, 2896, 85, 23, 2897, 85, 25, 2898, 85, 27, 2899, 85, 28, 2900  
DATA 85, 29, 2901, 85, 31, 2902, 85, 33, 2903, 85, 35, 2904, 85, 37, 2905, 85, 39, 2906, 85, 41, 2907, 85, 43, 2908, 85, 45, 2909, 85, 47, 2910  
DATA 85, 49, 2911, 85, 51, 2912, 85, 53, 2913, 85, 55, 2914, 85, 57, 2915, 85, 59, 2916, 85, 61, 2917, 86, 1, 2918, 86, 3, 2919, 86, 5, 2920  
DATA 86, 7, 2921, 86, 9, 2922, 86, 11, 2923, 86, 12, 2924, 86, 13, 2925, 86, 15, 2926, 86, 17, 2927, 86, 19, 2928, 86, 21, 2929, 86, 23, 2930  
DATA 86, 25, 2931, 86, 27, 2932, 87, 1, 2933, 87, 3, 2934, 87, 5, 2935, 87, 7, 2936, 87, 9, 2937, 87, 11, 2938, 87, 13, 2939, 87, 15, 2940  
DATA 87, 17, 2941, 87, 19, 2942, 87, 21, 2943, 87, 23, 2944, 87, 25, 2945, 87, 27, 2946, 87, 29, 2947, 87, 31, 2948, 87, 33, 2949, 87, 35, 2950  
DATA 87, 37, 2951, 87, 39, 2952, 87, 41, 2953, 87, 43, 2954, 87, 45, 2955, 87, 47, 2956, 87, 49, 2957, 87, 51, 2958, 87, 53, 2959, 87, 55, 2960  
DATA 87, 57, 2961, 88, 1, 2962, 88, 3, 2963, 88, 5, 2964, 88, 7, 2965, 88, 9, 2966, 88, 11, 2967, 88, 13, 2968, 88, 15, 2969, 88, 17, 2970  
DATA 88, 19, 2971, 88, 21, 2972, 88, 23, 2973, 88, 27, 2974, 88, 29, 2975, 88, 31, 2976, 88, 33, 2977, 88, 510, 2978, 91, 1, 2979, 91, 3, 2980  
DATA 91, 5, 2981, 91, 7, 2982, 91, 9, 2983, 91, 11, 2984, 91, 13, 2985, 91, 15, 2986, 91, 17, 2987, 91, 19, 2988, 91, 21, 2989, 91, 23, 2990  
DATA 91, 25, 2991, 91, 27, 2992, 91, 29, 2993, 91, 31, 2994, 91, 33, 2995, 91, 35, 2996, 91, 37, 2997, 91, 39, 2998, 91, 41, 2999, 91, 43, 3000  
DATA 91, 45, 3001, 91, 47, 3002, 91, 49, 3003, 91, 51, 3004, 91, 53, 3005, 91, 55, 3006, 91, 57, 3007, 91, 59, 3008, 91, 61, 3009, 91, 63, 3010  
DATA 91, 65, 3011, 91, 67, 3012, 91, 69, 3013, 91, 71, 3014, 91, 73, 3015, 91, 75, 3016, 91, 77, 3017, 92, 1, 3018, 92, 3, 3019, 92, 5, 3020

```
DATA 92, 7, 3021, 92, 9, 3022, 92, 11, 3023, 92, 13, 3024, 92, 15, 3025, 92, 17,
3026, 92, 19, 3027, 92, 21, 3028, 92, 23, 3029, 92, 25, 3030
DATA 92, 27, 3031, 92, 29, 3032, 92, 31, 3033, 92, 33, 3034, 92, 35, 3035, 92, 37,
3036, 92, 39, 3037, 92, 41, 3038, 92, 43, 3039, 92, 45, 3040
DATA 92, 47, 3041, 92, 49, 3042, 92, 51, 3043, 92, 53, 3044, 92, 55, 3045, 92, 57,
3046, 92, 59, 3047, 92, 61, 3048, 92, 63, 3049, 92, 65, 3050
DATA 92, 67, 3051, 92, 69, 3052, 92, 71, 3053, 93, 1, 3054, 93, 3, 3055, 93, 5,
3056, 93, 7, 3057, 93, 9, 3058, 93, 11, 3059, 93, 13, 3060
DATA 93, 15, 3061, 93, 17, 3062, 93, 19, 3063, 93, 21, 3064, 93, 23, 3065, 93, 25,
3066, 93, 27, 3067, 93, 29, 3068, 93, 31, 3069, 93, 33, 3070
DATA 93, 35, 3071, 93, 37, 3072, 93, 39, 3073, 93, 41, 3074, 93, 43, 3075, 93, 45,
3076, 93, 47, 3077, 93, 49, 3078, 93, 51, 3079, 93, 53, 3080
DATA 93, 55, 3081, 93, 57, 3082, 93, 59, 3083, 93, 61, 3084, 93, 63, 3085, 93, 65,
3086, 93, 67, 3087, 93, 69, 3088, 93, 71, 3089, 93, 73, 3090
DATA 93, 75, 3091, 93, 77, 3092, 93, 79, 3093, 93, 81, 3094, 93, 83, 3095, 93, 85,
3096, 93, 87, 3097, 93, 89, 3098, 93, 91, 3099, 93, 93, 3100
DATA 93, 95, 3101, 93, 97, 3102, 93, 99, 3103, 93, 101, 3104, 93, 103, 3105, 93,
105, 3106, 93, 107, 3107, 93, 109, 3108, 93, 111, 3109, 93, 113, 3110
DATA 93, 115, 3111
```

```
SUB MakeCountyConvert ()
```

```
    CONST COUNTYCOUNT = 3111
```

```
    DIM i AS INTEGER, j AS INTEGER
```

```
    DIM State AS INTEGER, County AS INTEGER, CountyNumber AS INTEGER
```

```
    FOR i = 1 TO MAXSTATE STEP 1
```

```
        FOR j = 1 TO MAXCOUNTY STEP 1
```

```
            CountyConvert(i, j) = 0
```

```
        NEXT j
```

```
    NEXT i
```

```
    FOR i = 1 TO COUNTYCOUNT STEP 1
```

```
        READ State, County, CountyNumber
```

```
        CountyConvert(State, County) = CountyNumber
```

```
    NEXT i
```

```
END SUB
```

FILENAME: SORTBIN1.FOR

C -----  
C  
C Program: sortbin1  
C -----  
C  
C Author(s): Steven Humphreys  
(505) 844-7223  
C  
C Sandia National Laboratories  
C Accident Analysis / Consequence Assessment Department  
C Albuquerque, NM 98715-0748  
C  
C Description:  
C  
C Usage:  
C  
C Inputs:  
C  
C Outputs:  
C  
C Variables:  
C  
C Calls:  
C  
C Assumptions:  
C  
C Version: 1.0  
C  
C Date: June 13, 1996  
C  
C Revision  
C History: None - initial version.  
C -----

program sortbin1  
integer BINMIN, BINMAX, BINCOUNT, FIRSTUNIT, RECORDSMAX  
parameter  
& (BINMIN = -26,  
& BINMAX = 33,  
& BINCOUNT = 30,  
& FIRSTUNIT = 10,  
& RECORDSMAX = 6930787  
& )  
character \* 17 FileName  
integer i, j, Block\_2Unit, Bin(BINMIN : BINMAX)  
integer LongitudeBin, Area  
integer \* 2 Longitude, Latitude, Population, CountyCode  
j = FIRSTUNIT  
do i = BINMIN, BINMAX, 2  
Bin(i) = j  
Bin(i + 1) = j

```

        write(*, *) i, i + 1, j
        j = j + 1

    end do

    j = BINMIN

    do i = FIRSTUNIT, BINCOUNT + FIRSTUNIT - 1, 1

        write(FileName, '(a, i2.2, a)') 'sort\block', i, '.bin'
        write(*, '(2i4, 1x, a)') Bin(j), Bin(j + 1), FileName
        open(unit = i, file = FileName, status = 'new',
              access = 'transparent', action = 'write')
        j = j + 2

    end do

    Block_2Unit = BINCOUNT + FIRSTUNIT
    FileName = 'block_2.bin'
    write(*, '(i8, 1x, a)') Block_2Unit, FileName
    open(unit = Block_2Unit, file = FileName, status = 'old',
          access = 'transparent', action = 'read')

    do i = 1, RECORDSMAX, 1

        read(Block_2Unit)
            Longitude, Latitude, Population, Area, CountyCode
        &

        LongitudeBin = (Longitude / 1000)
        if(LongitudeBin .gt. 0) then
            LongitudeBin = LongitudeBin + 1
        end if

        write(Bin(LongitudeBin))
            Longitude, Latitude, Population, Area, CountyCode
        &

        if((i .eq. 1) .or.
        &           (mod(i, 100000) .eq. 0) .or.
        &           (i .eq. RECORDSMAX)) then

            write(*, '(i8, i10, i9, i6, i9, i5)') i,
            &           (Longitude + 91993) * 1000,
            &           (Latitude + 16610) * 1000,
            &           Population, Area, CountyCode

        end if

    end do

    do i = FIRSTUNIT, BINCOUNT + FIRSTUNIT - 1, 1

        close(i)

    end do

    close(Block_2Unit)

end

```

C -----

FILENAME: SORTBIN2.FOR

```
C -----
C
C Program:      sortbin2
C
C
C Author(s):    Steven Humphreys
C                (505) 844-7223
C
C                Sandia National Laboratories
C                Accident Analysis / Consequence Assessment Department
C                Albuquerque, NM 98715-0748
C
C Description:
C
C Usage:
C
C Inputs:
C
C Outputs:
C
C Variables:
C
C Calls:
C
C Assumptions:
C
C Version:      1.0
C
C Date:         June 13, 1996
C
C Revision
C History:      None - initial version.
C
C -----
C
program sortbin2
    integer EOF, BINCOUNT, FIRSTUNIT, RECORDSMAX
    parameter
    (
    &     EOF      =      -1,
    &     BINCOUNT =      30,
    &     FIRSTUNIT =      10,
    &     RECORDSMAX = 553652
    & )
    Define the variables for determining the io status of the file
    operations we'll be performing. Note: this is not very portable
    code. It depends on specific io status codes to mean certain
    things, eg. -1 means an end of file. It also depends on the
    order of the most significant and least significant bytes of
    the two byte integer io_status. Finally it uses the two integer
    * size functions which are not standard f77 fortran.
    integer * 2 ioStatus
    integer * 1 ioStatusByte(2)
    equivalence (ioStatus, ioStatusByte)
```

```

character * 17 FileName
character * 80 ErrorMessage
integer i, j, Block_3Unit, RecordCount, Area(RECORDSMAX)
integer * 2 Longitude(RECORDSMAX), Latitude(RECORDSMAX)
integer * 2 Population(RECORDSMAX), CountyCode(RECORDSMAX)
integer LongLongitude(RECORDSMAX), Permutations(RECORDSMAX)

Block_3Unit = BINCOUNT + FIRSTUNIT
FileName = 'block_3.bin'
write(*, '(1x, 2a)') 'Opening File: ', FileName
open(unit = Block_3Unit, file = FileName, status = 'new',
      access = 'transparent', action = 'write')

& do i = BINCOUNT + FIRSTUNIT - 1, FIRSTUNIT, -1

  RecordCount = 0

  write(FileName, '(a, i2.2, a)') 'sort\block', i, '.bin'
  write(*, '(/, 1x, 2a)') 'Opening file: ', FileName
  open(unit = i, file = FileName, status = 'old',
        access = 'transparent', action = 'read')

& do j = 1, RECORDSMAX, 1

  read(i, iostat = ioStatus)
  &   Longitude(j), Latitude(j),
  &   Population(j), Area(j), CountyCode(j)

  if(ioStatusByte(1) .eq. 0) then
    RecordCount = RecordCount + 1
  else if(ioStatusByte(1) .eq. EOF) then
    exit
  else
    call iostat msg(ioStatus, ErrorMessage)
    write(*, '(7, 1x, a)') charnb(ErrorMessage)
    call error(char(13))
    call exit(ioStatus)
  end if

  if((j .eq. 1) .or. (mod(j, 100000) .eq. 0)) then
    write(*, '(i8, i10, i9, i6, i9, i5)') j,
    & (Longitude(j) + 91993) * 1000,
    & (Latitude(j) + 16610) * 1000,
    & Population(j), Area(j), CountyCode(j)
  end if

  end do

  close(i)

  write(*, '(1x, 2a)') 'Sorting file: ', FileName

C   Initiate the LongLongitude integer array. The algorithm
C   below allows us to sort on the Longitude (primary key), and
C   the latitude with only one pass through the IMSL sort
C   routine.

  do j = 1, RecordCount, 1
    LongLongitude(j) = Longitude(j) + 91993
    LongLongitude(j) = ior(ishft(LongLongitude(j), 16),
      & (Latitude(j) + 16610))
    Permutations(j) = j

```

```

    end do

    call svigp(RecordCount, LongLongitude,
    &           LongLongitude, Permutations)

    write(*, '(1x, 2a)') 'Saving file: ', FileName

    do j = RecordCount, 1, -1

        write(Block_3Unit)
        &           Longitude(Permutations(j)),
        &           Latitude(Permutations(j)),
        &           Population(Permutations(j)),
        &           Area(Permutations(j)),
        &           CountyCode(Permutations(j))

        if((j .eq. 1) .or. (mod(j, 100000) .eq. 0)) then
            write(*, '(i8, i10, i9, i6, i9, i5)')
            &           Permutations(j),
            &           (Longitude(Permutations(j)) + 91993) * 1000,
            &           (Latitude(Permutations(j)) + 16610) * 1000,
            &           Population(Permutations(j)),
            &           Area(Permutations(j)),
            &           CountyCode(Permutations(j))
        end if

        end do

    end do

    write(*, '(/, 1x, a)') 'Closing file: block_3.bin'
    close(Block_3Unit)

end

```

C -----

FILENAME: RMOVEDUP.FOR

```
C -----
C
C     Program:      rMOVEDUP
C
C -----
C
C     Author(s):    Steven Humphreys
C                 (505) 844-7223
C
C                 Sandia National Laboratories
C                 Accident Analysis / Consequence Assessment Department
C                 Albuquerque, NM 98715-0748
C
C     Description: Counts duplicate records (ie. equal longitudes and
C                 lattitudes) in the block_3.bin file.
C
C     Usage:
C
C     Inputs:
C
C     Outputs:
C
C     Variables:
C
C     Calls:
C
C     Assumptions:
C
C     Version:      1.0
C
C     Date:         June 13, 1996
C
C     Revision
C     History:     None - initial version.
C
C -----
```

```
program rMOVEDUP

    integer FIRSTUNIT, RECORDSMAX

    parameter
    (
        FIRSTUNIT = 10,
        RECORDSMAX = 6930787
    )

    character * 11 FileName
    integer i, Block_3Unit, Block_4Unit, DuplicateCount
    integer * 2 Longitude, Latitude, Population, CountyCode
    integer * 2 PresentLongitude, PresentLatitude
    integer * 2 PresentPopulation, MaxPopulation, PresentCountyCode
    integer Area, MaxArea, PresentArea

    Block_3Unit = FIRSTUNIT
    FileName = 'Block_3.bin'
    write(*, '(i8, 1x, a)') Block_3Unit, FileName
    open(unit = Block_3Unit, file = FileName, status = 'old',
         access = 'transparent', action = 'read')
```

```

Block_4Unit = FIRSTUNIT + 1
FileName = 'Block_4.bin'
write(*, '(i8, 1x, a)') Block_4Unit, FileName
open(unit = Block_4Unit, file = FileName, status = 'new',
      access = 'transparent', action = 'write')

DuplicateCount = 0
MaxPopulation = 0
MaxArea = 0

read(Block_3Unit)
  PresentLongitude, PresentLatitude,
  PresentPopulation, PresentArea, PresentCountyCode
&
do i = 2, RECORDSMAX, 1

  read(Block_3Unit)
    Longitude, Latitude, Population, Area, CountyCode

  if((PresentLongitude .eq. Longitude) .and.
     (PresentLatitude .eq. Latitude) .and.
     (PresentCountyCode .eq. CountyCode)) then

    DuplicateCount = DuplicateCount + 1
    PresentPopulation = PresentPopulation + Population
    PresentArea = PresentArea + Area

  else

    write(Block_4Unit)
      PresentLongitude, PresentLatitude,
      PresentPopulation, PresentArea, PresentCountyCode
&

    PresentLongitude = Longitude
    PresentLatitude = Latitude
    PresentPopulation = Population
    PresentArea = Area
    PresentCountyCode = CountyCode

  end if

  if(PresentPopulation .gt. MaxPopulation) then
    MaxPopulation = PresentPopulation
  end if

  if(PresentArea .gt. MaxArea) then
    MaxArea = PresentArea
  end if

  if((mod(i, 100000) .eq. 0) .or. (i .eq. RECORDSMAX)) then

    write(*, '(1x, 2(a, i8))')
      'Record          = ', i,
      '  Duplicate Count = ', DuplicateCount
    write(*, '(1x, 2(a, i8))')
      'Max Population = ', MaxPopulation,
      '  Max Area      = ', MaxArea
&
    end if

  end do

  write(Block_4Unit)

```

```
&      PresentLongitude, PresentLatitude,  
&      PresentPopulation, PresentArea, PresentCountyCode  
  
      close(Block_3Unit)  
      close(Block_4Unit)  
  
end
```

C -----

## APPENDIX H — TEST PROGRAM LISTINGS

FILENAME: TEST1.BAS

```
PresentCounty = County
StatePopulation = StatePopulation + Population
CountyPopulation = Population
StateArea = StateArea + Area
CountyArea = Area

ELSE

PresentState = State
PresentCounty = County
StatePopulation = StatePopulation + Population
CountyPopulation = CountyPopulation + Population
StateArea = StateArea + Area
CountyArea = CountyArea + Area

END IF

LOOP

END
```

FILENAME: TEST2.BAS

OPTION EXPLICIT

CONST MAXRECORDS = 6930787

DIM Longitude AS LONG, Latitude AS LONG  
DIM Population AS LONG, Area AS LONG  
DIM State AS STRING \* 1, County AS INTEGER

DIM ShortLongitude AS INTEGER, ShortLatitude AS INTEGER  
DIM ShortPopulation AS INTEGER, LongArea AS LONG  
DIM CountyCode AS INTEGER

DIM RecordNumber AS LONG

OPEN "block\_1.bin" FOR BINARY AS #1  
OPEN "block\_2.bin" FOR BINARY AS #2

GET #1, , Longitude  
GET #1, , Latitude  
GET #1, , Population  
GET #1, , Area  
GET #1, , State  
GET #1, , County

GET #2, , ShortLongitude  
GET #2, , ShortLatitude  
GET #2, , ShortPopulation  
GET #2, , LongArea  
GET #2, , CountyCode

PRINT USING "##### ##### ##### ##### ##### ##### ##### ## ##"; 1; Longitude;  
Latitude; Population; Area; ASC(State); County

PRINT USING "##### ##### ##### ##### ##### ##### ##### ##### ## ##"; 1;  
((ShortLongitude + 91993) \* 1000); ((ShortLatitude + 16610&) \* 1000);  
ShortPopulation; LongArea; CountyCode

PRINT

LPRINT USING "##### ##### ##### ##### ##### ##### ##### ## ##"; 1; Longitude;  
Latitude; Population; Area; ASC(State); County

LPRINT USING "##### ##### ##### ##### ##### ##### ##### ##### ## ##"; 1;  
((ShortLongitude + 91993) \* 1000); ((ShortLatitude + 16610&) \* 1000);  
ShortPopulation; LongArea; CountyCode

LPRINT

FOR RecordNumber = 100000 TO MAXRECORDS STEP 100000

GET #1, (((RecordNumber - 1) \* 19) + 1), Longitude  
GET #1, , Latitude  
GET #1, , Population  
GET #1, , Area  
GET #1, , State  
GET #1, , County

GET #2, (((RecordNumber - 1) \* 12) + 1), ShortLongitude  
GET #2, , ShortLatitude  
GET #2, , ShortPopulation  
GET #2, , LongArea  
GET #2, , CountyCode

```

    PRINT USING "##### ##### ##### ##### ##### ##### ##### ## ##"; RecordNumber;
Longitude; Latitude; Population; Area; ASC(State); County
    PRINT USING "##### ##### ##### ##### ##### ##### ##### ##### ## ##"; RecordNumber;
((ShortLongitude + 91993) * 1000); ((ShortLatitude + 16610&) * 1000);
ShortPopulation; LongArea; CountyCode
    PRINT

    LPRINT USING "##### ##### ##### ##### ##### ##### ##### ## ##"; RecordNumber;
Longitude; Latitude; Population; Area; ASC(State); County
    LPRINT USING "##### ##### ##### ##### ##### ##### ##### ##### ## ##"; RecordNumber;
((ShortLongitude + 91993) * 1000); ((ShortLatitude + 16610&) * 1000);
ShortPopulation; LongArea; CountyCode
    LPRINT

NEXT RecordNumber

    GET #1, (((MAXRECORDS - 1) * 19) + 1), Longitude
    GET #1, , Latitude
    GET #1, , Population
    GET #1, , Area
    GET #1, , State
    GET #1, , County

    GET #2, (((MAXRECORDS - 1) * 12) + 1), ShortLongitude
    GET #2, , ShortLatitude
    GET #2, , ShortPopulation
    GET #2, , LongArea
    GET #2, , CountyCode

    PRINT USING "##### ##### ##### ##### ##### ##### ##### ## ##"; MAXRECORDS;
Longitude; Latitude; Population; Area; ASC(State); County
    PRINT USING "##### ##### ##### ##### ##### ##### ##### ## ##"; MAXRECORDS;
((ShortLongitude + 91993) * 1000); ((ShortLatitude + 16610&) * 1000);
ShortPopulation; LongArea; CountyCode
    PRINT

    LPRINT USING "##### ##### ##### ##### ##### ##### ##### ## ##"; MAXRECORDS;
Longitude; Latitude; Population; Area; ASC(State); County
    LPRINT USING "##### ##### ##### ##### ##### ##### ##### ## ##"; MAXRECORDS;
((ShortLongitude + 91993) * 1000); ((ShortLatitude + 16610&) * 1000);
ShortPopulation; LongArea; CountyCode
    LPRINT

END

```

FILENAME: TEST3.BAS

OPTION EXPLICIT

DIM Longitude AS INTEGER, Lattitude AS INTEGER  
DIM Population AS INTEGER, Area AS LONG  
DIM CountyCode AS INTEGER

OPEN "block\_3.bin" FOR BINARY AS #1

DO

GET #1, , Longitude  
 GET #1, , Lattitude  
 GET #1, , Population  
 GET #1, , Area  
 GET #1, , CountyCode

IF (EOF(1) OR (INKEY\$ <> "")) THEN

EXIT DO

ELSE

PRINT USING "#####"; ((Longitude + 91993) \* 1000); ((Lattitude + 16610&) \* 1000); Population; Area; CountyCode  
 LPRINT USING "#####"; ((Longitude + 91993) \* 1000); ((Lattitude + 16610&) \* 1000); Population; Area; CountyCode

END IF

LOOP

END

FILENAME: TEST4.BAS

```
OPTION EXPLICIT

CONST MAXCOUNTY = 3111
CONST MAXRECORDS = 6660337

DIM i AS LONG
DIM Longitude AS INTEGER, Latitude AS INTEGER
DIM Population AS INTEGER, Area AS LONG, County AS INTEGER
DIM CountyPopulation(MAXCOUNTY) AS LONG, CountyArea(MAXCOUNTY) AS LONG

OPEN "block_4.bin" FOR BINARY AS #1
OPEN "testout4.txt" FOR OUTPUT AS #2 LEN = 32767

FOR i = 1 TO MAXCOUNTY STEP 1
    CountyPopulation(i) = 0
    CountyArea(i) = 0

NEXT i

FOR i = 1 TO MAXRECORDS STEP 1
    GET #1, , Longitude
    GET #1, , Latitude
    GET #1, , Population
    GET #1, , Area
    GET #1, , County

    CountyPopulation(County) = CountyPopulation(County) + Population
    CountyArea(County) = CountyArea(County) + Area

    IF ((i MOD 100000) = 0) THEN
        PRINT i, Longitude, Latitude, Population, Area, County
    END IF

NEXT i

FOR i = 1 TO MAXCOUNTY STEP 1
    PRINT USING "##### ##### ##### #####"; i; CountyPopulation(i); CountyArea(i)
    PRINT #2, USING "##### ##### ##### #####"; i; CountyPopulation(i);
    CountyArea(i)

NEXT i

END
```

**FILENAME: TEST5.BAS**

```
option explicit

dim HardCopy as integer
dim RecordNumber as long

dim NewLongitude as integer, NewLatitude as integer
dim NewPopulation as integer, NewArea as long
dim NewCountyCode as integer

dim OldLongitude as integer, OldLatitude as integer
dim OldPopulation as integer, OldArea as long
dim OldCountyCode as integer

HardCopy = 1
RecordNumber = 0&

OldLongitude = 32767
OldLatitude = 0
OldPopulation = 0
OldArea = 0&
OldCountyCode = 0

open "c:\secpop90\census\census90.dat" for binary as #1

do

    get #1, , NewLongitude
    get #1, , NewLatitude
    get #1, , NewPopulation
    get #1, , NewArea
    get #1, , NewCountyCode

    if (eof(1) or (inkey$ <> "")) then

        exit do

    end if

    RecordNumber = RecordNumber + 1

    if((RecordNumber = 1) or (RecordNumber = 6660337) or _
       ((RecordNumber mod 100000) = 0)) then

        print using "##### ##.## ##.## ##### ##### #####"; _
                    RecordNumber;
                    ((NewLongitude + 91993) / 1000); _
                    ((NewLatitude + 16610&) / 1000); _
                    NewPopulation; NewArea; NewCountyCode

        if(HardCopy) then

            lprint using "##### ##.## ##.## ##### ##### #####"; _
                        RecordNumber;
                        ((NewLongitude + 91993) / 1000); _
                        ((NewLatitude + 16610&) / 1000); _
                        NewPopulation; NewArea; NewCountyCode

        end if

    end if

end if
```

```

if(NewLongitude > OldLongitude) then

    print using "##### ##.## ##.## ##### ##### #####"; -
        RecordNumber - 1;
        ((OldLongitude + 91993) / 1000); -
        ((OldLatitude + 16610&) / 1000); -
        OldPopulation; OldArea; OldCountyCode
    print "This record is out of order!!!"
    print using "##### ##.## ##.## ##### ##### #####"; -
        RecordNumber;
        ((NewLongitude + 91993) / 1000); -
        ((NewLatitude + 16610&) / 1000); -
        NewPopulation; NewArea; NewCountyCode

    if(HardCopy) then

        lprint using "##### ##.## ##.## ##### ##### #####"; -
            RecordNumber - 1;
            ((OldLongitude + 91993) / 1000); -
            ((OldLatitude + 16610&) / 1000); -
            OldPopulation; OldArea; OldCountyCode
        lprint "This record is out of order!!!"
        lprint using "##### ##.## ##.## ##### ##### #####"; -
            RecordNumber;
            ((NewLongitude + 91993) / 1000); -
            ((NewLatitude + 16610&) / 1000); -
            NewPopulation; NewArea; NewCountyCode

    end if

    end if

    OldLongitude = NewLongitude
    OldLatitude = NewLatitude
    OldPopulation = NewPopulation
    OldArea = NewArea
    OldCountyCode = NewCountyCode

loop

end

```

**FILENAME: FIX1.BAS**

```
option explicit

dim HardCopy as integer
dim RecordNumber as long

dim NewLongitude as integer, NewLatitude as integer
dim NewPopulation as integer, NewArea as long
dim NewCountyCode as integer

dim OldLongitude as integer, OldLatitude as integer
dim OldPopulation as integer, OldArea as long
dim OldCountyCode as integer

HardCopy = 1
RecordNumber = 0&

OldLongitude = 32767
OldLatitude = 0
OldPopulation = 0
OldArea = 0&
OldCountyCode = 0

open "c:\secpop90\census\census90.dat" for binary as #1

do

    get #1, , NewLongitude
    get #1, , NewLatitude
    get #1, , NewPopulation
    get #1, , NewArea
    get #1, , NewCountyCode

    if (eof(1) or (inkey$ <> "")) then

        exit do

    end if

    RecordNumber = RecordNumber + 1

    if((RecordNumber = 1) or (RecordNumber = 6660337) or _
       ((RecordNumber mod 100000) = 0)) then

        print using "#####.### ##.## ##### ##### #####"; _
                    RecordNumber;
        ((NewLongitude + 91993) / 1000); -
        ((NewLatitude + 16610&) / 1000); -
        NewPopulation; NewArea; NewCountyCode

    if(HardCopy) then

        lprint using "#####.### ##.## ##### ##### #####"; _
                    RecordNumber;
        ((NewLongitude + 91993) / 1000); -
        ((NewLatitude + 16610&) / 1000); -
        NewPopulation; NewArea; NewCountyCode

    end if

end if
```

```

if((RecordNumber => 1396205&) and (RecordNumber <= 1396215&)) then
  print using "##### ##.## ##.## ##### ##### #####"; _
    RecordNumber;
    ((NewLongitude + 91993) / 1000); _
    ((NewLatitude + 16610&) / 1000); _
    NewPopulation; NewArea; NewCountyCode

  if(HardCopy) then
    lprint using "##### ##.## ##.## ##### ##### #####"; _
      RecordNumber;
      ((NewLongitude + 91993) / 1000); _
      ((NewLatitude + 16610&) / 1000); _
      NewPopulation; NewArea; NewCountyCode

  end if
end if

if((RecordNumber => 1675345&) and (RecordNumber <= 1675355&)) then
  print using "##### ##.## ##.## ##### ##### #####"; _
    RecordNumber;
    ((NewLongitude + 91993) / 1000); _
    ((NewLatitude + 16610&) / 1000); _
    NewPopulation; NewArea; NewCountyCode

  if(HardCopy) then
    lprint using "##### ##.## ##.## ##### ##### #####"; _
      RecordNumber;
      ((NewLongitude + 91993) / 1000); _
      ((NewLatitude + 16610&) / 1000); _
      NewPopulation; NewArea; NewCountyCode

  end if
end if

if((RecordNumber => 1774660&) and (RecordNumber <= 1774670&)) then
  print using "##### ##.## ##.## ##### ##### #####"; _
    RecordNumber;
    ((NewLongitude + 91993) / 1000); _
    ((NewLatitude + 16610&) / 1000); _
    NewPopulation; NewArea; NewCountyCode

  if(HardCopy) then
    lprint using "##### ##.## ##.## ##### ##### #####"; _
      RecordNumber;
      ((NewLongitude + 91993) / 1000); _
      ((NewLatitude + 16610&) / 1000); _
      NewPopulation; NewArea; NewCountyCode

  end if
end if

OldLongitude = NewLongitude
OldLatitude = NewLatitude

```

```
OldPopulation = NewPopulation
OldArea = NewArea
OldCountyCode = NewCountyCode
```

```
loop
```

```
end
```

**FILENAME: FIX2.BAS**

```
option explicit

dim HardCopy as integer
dim RecordNumber as long

dim Longitude as integer, Latitude as integer
dim Population as integer, Area as long
dim CountyCode as integer

HardCopy = 1

open "c:\secpop90\census\census90.dat" for binary access read as #1
open "temp.dat" for binary as #2

for RecordNumber = 1& to 1396211& step 1

    get #1, , Longitude
    get #1, , Latitude
    get #1, , Population
    get #1, , Area
    get #1, , CountyCode

    put #2, , Longitude
    put #2, , Latitude
    put #2, , Population
    put #2, , Area
    put #2, , CountyCode

    if((RecordNumber = 1) or (RecordNumber = 6660337) or _
       ((RecordNumber mod 100000) = 0)) then

        print using "##### ##.## ##.## ##### ##### #####"; _
        RecordNumber;
        ((Longitude + 91993) / 1000); -
        ((Latitude + 16610&) / 1000); -
        Population; Area; CountyCode

        if(HardCopy) then

            lprint using "##### ##.## ##.## ##### ##### #####"; _
            RecordNumber;
            ((Longitude + 91993) / 1000); -
            ((Latitude + 16610&) / 1000); -
            Population; Area; CountyCode

        end if

    end if

next RecordNumber

for RecordNumber = 1396212& to 1675348& step 1

    get #1, , Longitude
    get #1, , Latitude
    get #1, , Population
    get #1, , Area
    get #1, , CountyCode

next RecordNumber
```

```

for RecordNumber = 1675349& to 1774662& step 1

  get #1, , Longitude
  get #1, , Lattitude
  get #1, , Population
  get #1, , Area
  get #1, , CountyCode

  put #2, , Longitude
  put #2, , Lattitude
  put #2, , Population
  put #2, , Area
  put #2, , CountyCode

  if((RecordNumber = 1) or (RecordNumber = 6660337) or _
     ((RecordNumber mod 100000) = 0)) then

    print using "##### ##.## ##.## ##### ##### ##### #####"; _
      RecordNumber;
      ((Longitude + 91993) / 1000);-
      ((Lattitude + 16610&) / 1000);-
      Population; Area; CountyCode

    if(HardCopy) then

      lprint using "##### ##.## ##.## ##### ##### ##### #####"; _
        RecordNumber;
        ((Longitude + 91993) / 1000);-
        ((Lattitude + 16610&) / 1000);-
        Population; Area; CountyCode

    end if

  end if

next RecordNumber

close(1)

open "c:\secpop90\census\census90.dat" for binary access read as #1

for RecordNumber = 1& to 1396211& step 1

  get #1, , Longitude
  get #1, , Lattitude
  get #1, , Population
  get #1, , Area
  get #1, , CountyCode

next RecordNumber

for RecordNumber = 1396212& to 1675348& step 1

  get #1, , Longitude
  get #1, , Lattitude
  get #1, , Population
  get #1, , Area
  get #1, , CountyCode

  put #2, , Longitude
  put #2, , Lattitude
  put #2, , Population

```

```

put #2, , Area
put #2, , CountyCode

if((RecordNumber = 1) or (RecordNumber = 6660337) or _
  ((RecordNumber mod 100000) = 0)) then

  print using "##### ##.## ##.## ##### ##### ##### ##"; _
    RecordNumber;
    ((Longitude + 91993) / 1000); -
    ((Latitude + 16610&) / 1000); -
    Population; Area; CountyCode

  if(HardCopy) then

    lprint using "##### ##.## ##.## ##### ##### ##### ##"; _
      RecordNumber;
      ((Longitude + 91993) / 1000); -
      ((Latitude + 16610&) / 1000); -
      Population; Area; CountyCode

  end if

end if

next RecordNumber

for RecordNumber = 1675349& to 1774662& step 1

  get #1, , Longitude
  get #1, , Latitude
  get #1, , Population
  get #1, , Area
  get #1, , CountyCode

next RecordNumber

for RecordNumber = 1774663& to 6660337& step 1

  get #1, , Longitude
  get #1, , Latitude
  get #1, , Population
  get #1, , Area
  get #1, , CountyCode

  put #2, , Longitude
  put #2, , Latitude
  put #2, , Population
  put #2, , Area
  put #2, , CountyCode

  if((RecordNumber = 1) or (RecordNumber = 6660337) or _
    ((RecordNumber mod 100000) = 0)) then

    print using "##### ##.## ##.## ##### ##### ##### ##"; _
      RecordNumber;
      ((Longitude + 91993) / 1000); -
      ((Latitude + 16610&) / 1000); -
      Population; Area; CountyCode

    if(HardCopy) then

      lprint using "##### ##.## ##.## ##### ##### ##### ##"; _
        RecordNumber; -

```

```
((Longitude + 91993) / 1000); _  
((Latitude + 16610&) / 1000); _  
Population; Area; CountyCode  
end if  
end if  
next RecordNumber  
close(1)  
close(2)  
end
```



## APPENDIX I—SECPOP90 PROGRAM LISTINGS

FILENAME: CALCCODE.BAS

OPTION EXPLICIT

'\$INCLUDE: 'SECPOP90.BI'

' THIS SUBROUTINE ALLOCATES THE 1990 CENSUS DATA TO A MACCS GRID FOR  
' MULTIPLE SITES BASED ON INTERACTIVE COMMUNICATION WITH THE USER.  
' EACH SITE LOCATION IS DEFINED BY THE NUMBER OF DEGREES, MINUTES,  
' AND SECONDS OF LATITUDE AND LONGITUDE. THE MACCS GRID IS BASED  
' ON 16 RADIAL DIRECTIONS COUNTED CLOCKWISE FROM DUE NORTH WITH  
' DIRECTION 1 HAVING DUE NORTH AS ITS AXIS OF SYMMETRY.

' GLOSSARY OF VARIABLES:

' AMINLA = LATITUDE OF TROPIC OF CANCER  
' BNDRYE = LONGITUDE OF EASTERNMOST POINT ON GRID  
' BNDRYN = LATITUDE OF NORTHERNMOST POINT ON GRID  
' BNDRYS = LATITUDE OF SOUTHERNMOST POINT ON GRID  
' BNDRYW = LONGITUDE OF WESTERNMOST POINT ON GRID  
' DELLAT = DIFFERENCE IN LATITUDE FROM SITE TO GRID BOUNDARIES  
' DELLON = DIFFERENCE IN LONGITUDE FROM SITE TO GRID BOUNDARIES  
' DISMAX = MAXIMUM RADIAL DISTANCE FOR GRID (KM)  
' DPDLAT = DISTANCE PER DEGREE LATITUDE (KM)  
' DPDLON = DISTANCE PER DEGREE LONGITUDE (KM)  
' NUMDIR = NUMBER OF RADIAL DIRECTIONS IN GRID  
' NUMRAD = NUMBER OF RADIAL DISTANCES DESIGNATED  
' OFFSET = RADIANS TO END OF FIRST RADIAL DIRECTION  
' RADDIS = ARRAY OF RADIAL DISTANCES IN MACCS GRID  
' SCALE = POPULATION SCALING FACTOR (MULTIPLIER) FOR SITE  
' SDPDLA = DISTANCE PER DEGREE LATITUDE AT THE SITE  
' SDPDLO = DISTANCE PER DEGREE LONGITUDE AT THE SITE  
' SITID\$ = SITE IDENTIFIER  
' SLAT = SITE LATITUDE IN DECIMAL DEGREES  
' SLON = SITE LONGITUDE IN DECIMAL DEGREES  
' sector\_population = ARRAY OF POPULATION FOUND IN EACH GRID ELEMENT  
' XLON = LONGITUDE OF POINT READ FROM CENSUS DATA FILE  
' YLAT = LATITUDE OF POINT READ FROM CENSUS DATA FILE

' EXTERNALS:

' GETDIR = DETERMINES THE RADIAL DIRECTION OF A POINT (XLON,YLAT)  
' FROM THE SITE AT (SLON,SLAT)  
' GETDIS = DETERMINES THE DISTANCE (KM) PER DEGREE FOR BOTH  
' LATITUDE AND LONGITUDE AT A GIVEN LATITUDE  
' GETRAD = DETERMINES THE RADIAL SECTOR IN WHICH POINT (XLON,  
' YLAT) LIES RELATIVE TO THE SITE (SLAT,SLON)  
' POINTR = FINDS THE FIRST RECORD IN THE SORTED DATA FILE IN  
' WHICH THE POINT LIES ON OR TO THE EAST OF THE WESTERN  
' BOUNDARY OF THE GRID

FUNCTION DIST (X, Y)

    DIST = SQR(X ^ 2! + Y ^ 2!)

END FUNCTION

SUB GETDIR (ylat, idir AS INTEGER)

' THIS SUBROUTINE RETURNS THE DIRECTIONAL ELEMENT IN WHICH POINT  
' (XLON,YLAT) LIES

```

' GLOSSARY OF VARIABLES:
' ANGHI = ANGLE OF RIGHT SIDE OF THE RADIAL DIRECTION CONSIDERED
' ANGLO = ANGLE OF LEFT SIDE OF THE RADIAL DIRECTION CONSIDERED
' IDIR = RADIAL DIRECTION IN WHICH POINT (XLON,YLAT) LIES
' OFFSET = ANGLE OF RIGHT SIDE OF RADIAL DIRECTION 1
' SLAT = LATITUDE OF SITE
' SLON = LONGITUDE OF SITE
' THETA = COUNTERCLOCKWISE ANGLE FROM NORTH AT WHICH POINT (XLON,YLAT)
' LIES RELATIVE TO THE SITE
' XLON = LONGITUDE OF THE SITE
' YLAT = LATITUDE OF THE SITE

DIM OFFSET, AVDPD, DX, DY, THETA, K, j, L
DIM ANGLO, ANGHI, i

OFFSET = pi / 16!

idir = 0
IF (SLAT = ylat) THEN
    idir = 5
    IF (SLON <= xlon) THEN
        idir = idir + 8
    END IF
ELSE
    AVDPD = (SDPDLA + DPDLAT) / 2!
    DX = DPDLON * (SLON - xlom)
    DY = AVDPD * (ylat - SLAT)
    THETA = ATN(DX / DY)
    IF ((ABS(THETA) < OFFSET) OR (THETA = OFFSET)) THEN
        idir = 1
        IF (ylat < SLAT) THEN
            idir = idir + 8
        END IF
    ELSE
        j = 1
        FOR i = 1 TO 3
            K = i * 2 + 1
            L = i * 2 - 1
            j = j + 1
            ANGLO = OFFSET * L
            ANGHI = OFFSET * K
            IF ((ANGLO < ABS(THETA)) AND (ABS(THETA) < ANGHI)) THEN
                IF (THETA > 0!) THEN
                    idir = j
                    IF (ylat < SLAT) THEN
                        idir = idir + 8
                    END IF
                ELSE
                    idir = 18 - j
                    IF (ylat < SLAT) THEN
                        idir = idir - 8
                    END IF
                END IF
            END IF
        NEXT i
    END IF
END IF

IF (idir = 0) THEN
    IF (THETA > 0!) THEN
        idir = 5

```

```

        IF (xlon > SLON) THEN
            idir = idir + 8
        END IF
    ELSE
        idir = 13
        IF (xlon < SLON) THEN
            idir = idir - 8
        END IF
    END IF
END IF

END SUB

SUB GETDIS (Y AS SINGLE)

' THIS SUBROUTINE RETURNS THE DISTANCE PER DEGREE LATITUDE AND
' DISTANCE PER DEGREE LONGITUDE AT THE GEODETIC LATITUDE (SLAT)

' GLOSSARY OF VARIABLES:
' COSGC = COSINE OF GEOCENTRIC ANGLE
' COS2GC = SQUARE OF COSGC
' DGTORD = CONVERSION FACTOR FOR DEGREES TO RADIANS
' DPDLAT = DISTANCE PER DEGREE LATITUDE AT SLAT (KM)
' DPDLON = DISTANCE PER DEGREE LONGITUDE AT SLON (KM)
' GDRLA = GEODETIC LATITUDE IN RADIANS

' QLAT = GEODETIC LATITUDE OF POINT CONSIDERED
' RE2 = SQUARE OF EQUATORIAL RADIUS OF EARTH
' RGD = RADIUS OF GEODETIC LATITUDE CIRCLE (KM)
' RLONG = RADIUS OF LONGITUDE "CIRCLE" (KM)
' RP2 = SQUARE OF POLAR RADIUS OF EARTH
' RPROD = PRODUCT OF EQUATORIAL AND POLAR RADII
' SINGC = SINE OF GEOCENTRIC ANGLE
' SIN2GC = SQUARE OF SINGC
' THGC = GEOCENTRIC ANGLE CORRESPONDING TO THE GEODETIC
' LATITUDE

DIM RE, RE2, RP, RP2, RRATIO, RPROD, GDRLA, THGC, SINGC, SIN2GC, COSGC, COS2GC
DIM RGC, RGD, RLONG

' DEFINE CONSTANTS

RE = 6378.137
RE2 = RE ^ 2!
RP = 6356.7523142#
RP2 = RP ^ 2!
RRATIO = RP2 / RE2
RPROD = RE * RP

' FIND THE GEODETIC LATITUDE IN RADIANS

GDRLA = Y * DGTORD

' FIND THE GEODETIC LATITUDE IN THE MASTER COORDINATE SYSTEM

THGC = ATN(RRATIO * TAN(GDRLA))
SINGC = SIN(THGC)
SIN2GC = SINGC ^ 2!
COSGC = COS(THGC)
COS2GC = COSGC ^ 2!

' FIND THE GEOCENTRIC RADIUS (KM)

```

```

RGC = RPROD / SQR(RE2 * SIN2GC + RP2 * COS2GC)

' FIND THE RADIUS OF THE LONGITUDE "CIRCLE"

RLONG = RGC * COSGC

' FIND THE GEODETIC RADIUS

RGD = RLONG * (TAN(THGC) / SIN(GDRLA))

' FIND THE DISTANCE PER DEGREE LATITUDE AND LONGITUDE (KM)

DPDLAT = RGD * DGTORD
DPDLON = RLONG * DGTORD

END SUB

SUB GETRAD (ylat, NUMRAD AS INTEGER, irad AS INTEGER)

' THIS SUBROUTINE GETS THE RADIAL SPATIAL ELEMENT IN WHICH THE
' POINT (XLON,YLAT) LIES

' GLOSSARY OF VARIABLES:
' AVDPD = AVERAGE DISTANCE PER DEGREE LATITUDE BETWEEN THE SITE
'          AND A POINT AT LATITUDE YLAT
' DELX = DIFFERENCE IN LONGITUDE BETWEEN A POINT AT (XLON,
'          YLAT) AND THE SITE
' DELY = DIFFERENCE IN LATITUDE BETWEEN A POINT AT (XLON,YLAT)
'          AND THE SITE
' DISMAX = RADIUS OF THE FULL MACCS GRID
' DISTST = DISTANCE FROM POINT (XLON,YLAT) TO THE SITE
' IRAD = THE RADIAL SECTOR OF THE MACCS GRID IN WHICH POINT
'          (XLON, YLAT) LIES
' RADDIS = ARRAY OF RADIAL DISTANCES OF THE MACCS GRID BEING USED
' SLAT = LATITUDE OF THE SITE
' SLON = LONGITUDE OF THE SITE

DIM DELX, AVDPD, DELY, DISTST, JRAD

DELX = ABS(SLON - xlon) * DPDLON
AVDPD = (SDPDLA + DPDLAT) / 2!
DELY = ABS(SLAT - ylat) * AVDPD

DISTST = DIST(DELX, DELY)

irad = 0

IF (DISTST <= DISMAX) THEN
    IF (DISTST <= RADDIS(1)) THEN
        irad = 1
    ELSEIF (NUMRAD >= 2) THEN
        FOR JRAD = 2 TO NUMRAD
            IF ((DISTST > RADDIS(JRAD - 1)) AND (DISTST <= RADDIS(JRAD))) THEN
                irad = JRAD
                EXIT FOR
            END IF
        NEXT JRAD
    END IF
END IF

END SUB

```

```

SUB pointr (BYVAL BNDRYW, census_file AS LONG, ylat, ipop AS INTEGER, area AS LONG,
county_code AS INTEGER, ibyte AS LONG)

' THIS SUBROUTINE FINDS THE FIRST RECORD IN THE CENSUS DATA FILE
' WHICH IS AT THE WESTERN BOUNDARY OF THE GRID

' GLOSSARY OF VARIABLES:
'   ALAT   = POINT LATITUDE IN DECIMAL NOTATION
'   ALON   = POINT LONGITUDE IN DECIMAL NOTATION
'   BNDRYW = LONGITUDE OF WESTERN BOUNDARY OF GRID
'   ILAT   = CODED INTEGER REPRESENTATION OF LATITUDE OF POINT
'             IN CENSUS DATA FILE
'   ILOM   = CODED INTEGER REPRESENTATION OF LONGITUDE OF POINT
'             IN CENSUS DATA FILE
'   ILONW  = CODED INTEGER REPRESENTATION OF WESTERN BOUNDARY
'             LONGITUDE OF GRID
'   XLON   = LONGITUDE OF FIRST CENSUS DATA POINT LOCATED ON OR
'             WITHIN THE MACCS GRID
'   YLAT   = LATITUDE OF FIRST CENSUS DATA POINT LOCATED ON OR
'             WITHIN THE MACCS GRID

CONST NUMREC = number_of_records

'*****ROUTINE TO READ FROM RANDOM ACCESS BINARY FILE *****
'*****ROUTINE TO READ FROM RANDOM ACCESS BINARY FILE *****
'*****ROUTINE TO READ FROM RANDOM ACCESS BINARY FILE *****

DIM MIDREC AS LONG, NBYTE AS LONG
DIM IBEG AS LONG, IEND AS LONG
DIM ILONW AS LONG, ilon, A$, JLON, ilat, CEND$

ILONW = BNDRYW * 1000 - longitude_offset
GET #census_file, 1, rec
ilon = CVI(MID$(rec, 1, 2))
IF (ilon > ILONW) THEN
    IBEG = 1
    IEND = NUMREC
    ibyte = 1
ELSE
    ibyte = 1
    GOTO process
END IF

ReadRecord:
    MIDREC = INT((IBEG + IEND) / 2)
    NBYTE = ibyte
    ibyte = ((MIDREC - 1) * record_length) + 1
    IF (NBYTE <> ibyte) THEN
        GET #census_file, ibyte, rec
        ilon = CVI(MID$(rec, 1, 2))
        IF (ilon > ILONW) THEN
            IBEG = MIDREC
            GOTO ReadRecord
        ELSEIF (ilon = ILONW) THEN
            GOTO FindFirst
        ELSEIF (ilon < ILONW) THEN
            A$ = rec
            ibyte = ibyte - record_length
            GET #census_file, ibyte, rec
            JLON = CVI(MID$(rec, 1, 2))
            IF (JLON < ILONW) THEN
                IEND = MIDREC

```

```

        GOTO ReadRecord
    ELSEIF (JLON = ILONW) THEN
        GOTO FindFirst
    ELSEIF (JLON > ILONW) THEN
        ibyte = ibyte + record_length
        rec = A$
        GOTO process
    END IF
END IF
ELSE
    MSGBOX "Site not within Continental US.", MB_OK, "Error"
    EXIT SUB
END IF

' Find the first point inside the western boundary

FindFirst:

    ibyte = ibyte - record_length
    A$ = rec
    GET #census_file, ibyte, rec
    ilon = CVI(MID$(rec, 1, 2))
    IF (ilon = ILONW) THEN
        GOTO FindFirst
    ELSE
        IF (ilon > ILONW) THEN
            ibyte = ibyte + record_length
            rec = A$
            GOTO process
        END IF
    END IF

' When find the first point process the information

process:

    ilon = CVI(MID$(rec, 1, 2))
    ilat = CVI(MID$(rec, 3, 2))
    ipop = CVI(MID$(rec, 5, 2))
    area = CVL(MID$(rec, 7, 4))
    county_code = CVI(MID$(rec, 11, 2))

    xlon = (ilon + longitude_offset) / 1000!
    ylat = (ilat + latitude_offset) / 1000!

END SUB

SUB popcalc ()

    DIM tempstring AS STRING
    DIM census_file AS LONG, county_file AS LONG, debug_file AS LONG
    DIM A$, NUMLIN AS INTEGER, INUM AS INTEGER, B$, JBEGL AS INTEGER
    DIM JEND AS INTEGER, NLEFT AS INTEGER, C$
    DIM L1$, L2$, L3$, L4$, L5$, L6$, L7$, LA1 AS INTEGER, LA2 AS INTEGER
    DIM LA3 AS INTEGER, LO1 AS INTEGER, LO2 AS INTEGER, LO3 AS INTEGER
    DIM D$, DELLON, DELLAT, ilon AS INTEGER, ilat AS INTEGER
    DIM NP1, NP2, NP3, NP4, NP5, NP6, NP7, NP8, CEND$
    DIM ILIN AS INTEGER, i AS INTEGER, j AS INTEGER, m AS INTEGER
    DIM events_open AS INTEGER, average_counter AS INTEGER
    DIM start_file_pos AS LONG, end_file_pos AS LONG
    DIM num_records AS LONG, num_processed AS LONG, census_records AS LONG
    DIM hours AS INTEGER, minutes AS INTEGER, seconds AS SINGLE
    DIM time_begin AS LONG, time_end AS LONG, total_time AS LONG

```

```

DIM prev_seconds AS SINGLE

'Open census data file, located at the position specified
'by the user in the setup form

tempstring = frmSetup.txtData_Path.text + "\CENSUS90.DAT"
census_file = FREEFILE
OPEN tempstring FOR BINARY ACCESS READ AS #census_file LEN = 32767

'Initialize the county data arrays to their startup values
'These arrays contain various types of county economic
'data and state abbreviations.

frmCalculate.lblStatus.caption = "Reading county information."
ipop = 0
tempstring = frmSetup.txtData_Path.text + "\COUNTY90.DAT"
county_file = FREEFILE
OPEN tempstring FOR INPUT ACCESS READ AS #county_file

'debug_file = FREEFILE
'OPEN "temp.txt" FOR OUTPUT ACCESS WRITE AS #debug_file

'Read in the header line. County_state(0) is used as a dummy variable.

LINE INPUT #county_file, county_state(0)

'Read in the rest of the file.

FOR i = 1 TO number_of_counties STEP 1
    INPUT #county_file, county_code
    county_state(i) = INPUT$(2, #county_file)
    county_name(i) = INPUT$(4, #county_file)
    county_name(i) = INPUT$(max_county_name_length, #county_file)
    INPUT #county_file, county_frclnd(i)
    INPUT #county_file, county_frmfrc(i)
    INPUT #county_file, county_dpf(i)
    INPUT #county_file, county_asfp(i)
    INPUT #county_file, county_vfrm(i)
    INPUT #county_file, county_vnfrm(i)
    'Skip over the notes.
    LINE INPUT #county_file, county_state(0)
    'PRINT #debug_file, USING "#####"; county_code;
    'PRINT #debug_file, USING "& "; county_state(i);
    'PRINT #debug_file, USING "& "; county_name(i);
    'PRINT #debug_file, USING "#.#####"; county_frclnd(i);
    'PRINT #debug_file, USING "#.#####"; county_frmfrc(i);
    'PRINT #debug_file, USING "#.#####"; county_dpf(i);
    'PRINT #debug_file, USING "#####"; county_asfp(i);
    'PRINT #debug_file, USING "#####"; county_vfrm(i);
    'PRINT #debug_file, USING "#####"; county_vnfrm(i)
NEXT i

CLOSE (county_file)
'CLOSE (debug_file)

'Set degrees to radians conversion

DGTORD = pi / 180!

'Set the latitude of the tropic of cancer

AMINLA = 23.45

```

```

'Initialize RADDIS array to the values in the radial distance array so
'that the values can be changed locally without affecting other modules.

FOR i = 1 TO number_of_radii
    IF frmProblem_Data.optKilometers.value = TRUE THEN
        RADDIS(i) = radial_distance(i)
    ELSE
        RADDIS(i) = radial_distance(i) * miles_to_kilometers
    END IF
NEXT i

'Initialize the population and land fraction sector arrays and economics
'region arrays to zero.

FOR i = 1 TO number_of_segments STEP 1
    FOR j = 1 TO number_of_radii STEP 1
        sector_population(i, j) = 0
        sector_area(i, j) = 0
        sector_frclnd(i, j) = 0
    NEXT j
NEXT i

FOR i = 0 TO number_econ_regions
    econ_data(i).region_area = 0
    econ_data(i).region_frmfrc = 0
    econ_data(i).region_dpf = 0
    econ_data(i).region_asfp = 0
    econ_data(i).region_vfrm = 0
    econ_data(i).region_vnfrm = 0
NEXT i

'Set local constants for number of radii and distance of farthest radii.

NUMRAD = number_of_radii
DISMAX = RADDIS(NUMRAD)

```

ProcessSite:

```

'Reading interactive input.

SITID$ = frmSite_Data.txtSite_Name.text
L1$ = frmSite_Data.txtLatitude_Degrees.text
L2$ = frmSite_Data.txtLatitude_Minutes.text
L3$ = frmSite_Data.txtLatitude_Seconds.text
L4$ = frmSite_Data.txtLongitude_Degrees.text
L5$ = frmSite_Data.txtLongitude_Minutes.text
L6$ = frmSite_Data.txtLongitude_Seconds.text
L7$ = frmProblem_Data.txtPopulation_multiplier.text

'Convert interactive input strings into numeric values.

LA1 = INT(VAL(L1$))
LA2 = INT(VAL(L2$))
LA3 = INT(VAL(L3$))
LO1 = INT(VAL(L4$))
LO2 = INT(VAL(L5$))
LO3 = INT(VAL(L6$))
SCALE = VAL(L7$)

'Convert Degree/Minute/Second designation to decimal degrees.

SLAT = VAL(L1$) + VAL(L2$) / 60! + VAL(L3$) / 3600!
SLON = VAL(L4$) + VAL(L5$) / 60! + VAL(L6$) / 3600!

```

```

'Find the longitude-latitude boundaries of grid.
'Set distance per degree latitude for site.

CALL GETDIS(SLAT)
SDPDLA = DPDLAT

'Set eastern and western boundaries. These boundaries determine which
'records are searched in the census file.

DELLON = DISMAX / DPDLON
BNDRYW = INT((SLON + DELLON) * 1000 + .5) / 1000
BNDRYE = INT((SLON - DELLON) * 1000 + .5) / 1000

'Find distance change per degree latitude at some minimum latitude
'(AMINLA).

CALL GETDIS(AMINLA)

'Set northern and southern boundaries. These boundaries do not affect
'which records are searched, but do determine whether or not a point
'is inside of the circle being analyzed.

DELLAT = DISMAX / SDPDLA
BNDRYN = INT((SLAT + DELLAT) * 1000 + .5) / 1000
DELLAT = DISMAX / DPDLAT
BNDRYS = INT((SLAT - DELLAT) * 1000 + .5) / 1000

'Find the last record in the file with the longitude within the
'longitudinal boundaries of the grid. This record is used to
'estimate the run time for the problem.

xlon = 0!
ylat = 0!
area = 0&
county_code = 0
CALL pointr(BYVAL BNDRYE, census_file, ylat, ipop, area, county_code, ibyte)
end_file_pos = ibyte
IF (xlon = 0!) THEN
    end_file_pos = LOF(census_file)
END IF

frmCalculate.lblStatus.caption = "Searching for first census record."

ipop = 0
'Find the first record in the file with the longitude within the
'longitudinal boundaries of the grid.

ipop = 0
xlon = 0!
ylat = 0!
area = 0&
county_code = 0
CALL pointr(BYVAL BNDRYW, census_file, ylat, ipop, area, county_code, ibyte)
IF (xlon = 0!) THEN
    GOTO finishup
END IF
start_file_pos = ibyte

'Determine number of records to be scanned in census file (this
'number is used in determining estimated run time for a problem).

num_records = ((end_file_pos - start_file_pos) / record_length)

```

```

'Verify that the starting position is within the specified
'longitude and latitude. If so, read in and process the data
'from the first specified record in the census database.

idir = 0
irad = 0

IF ((xlon <= BNDRYW) AND (xlon >= BNDRYE)) THEN
    IF ((ylat <= BNDRYN) AND (ylat >= BNDRYS)) THEN
        'Determine the distance between the specified site and the first
        'location in the census database.

        CALL GETDIS(ylat)
        CALL GETRAD(ylat, NUMRAD, irad)

        'If the point is outside of the outermost radius of the circle,
        'read in and process the next record ...

        IF (irad = 0) THEN
            GOTO Continuel

        ELSE
            'Otherwise, determine in which sector the block lies.

            CALL GETDIR(ylat, idir)

            'Set the population of the sector to the population of the
            'block.

            sector_population(idir, irad) = ipop * SCALE
            sector_area(idir, irad) = area
            sector_frclnd(idir, irad) = area * county_frclnd(county_code)

            'Set the regional economic data by setting each of the
            'appropriate elements in the user defined data type.

            econ_data(region_index(irad, idir - 1)).region_area = area
            econ_data(region_index(irad, idir - 1)).region_frmfrc = area * *
            county_frmfrc(county_code)
            econ_data(region_index(irad, idir - 1)).region_dpf = area * *
            county_dpf(county_code)
            econ_data(region_index(irad, idir - 1)).region_asfp = area * *
            county_asfp(county_code)
            econ_data(region_index(irad, idir - 1)).region_vfrm = area * *
            county_vfrm(county_code)
            econ_data(region_index(irad, idir - 1)).region_vnfrm = area * *
            county_vnfrm(county_code)

            'Set number of census records processed thus far.

            census_records = 1

        END IF

    END IF

END IF

```

Continuel:

```
'Update labels on calculate form, and begin timer to estimate
'the run time of the calculation

frmCalculate.lblStatus.caption = "Processing census records."
time_begin = TIMER

'Read each subsequent record in the data file until the first record
'with a longitude outside of the longitudinal boundaries

readit:

'If end of file is reached, finish the economic calculations and exit.

IF (EOF(census_file)) THEN
    GOTO finishup
END IF

'Change to get statement with binary file, and Go to the next record.

ibyte = ibyte + record_length

'Read the data from that record, split it into its constituent pieces,
'and convert those pieces from file format to program format. This
'portion of code is done by the pointr function for the first record.

GET #census_file, ibyte, rec
ilon = CVI(MID$(rec, 1, 2))
ilat = CVI(MID$(rec, 3, 2))
ipop = CVI(MID$(rec, 5, 2))
area = CVL(MID$(rec, 7, 4))
county_code = CVI(MID$(rec, 11, 2))
xlon = (ilon + longitude_offset) / 1000!
ylat = (ilat + latitude_offset) / 1000!

'Check and see if the point is within the lat and long boundaries of
'the circle.

IF ((xlon <= BNDRYW) AND (xlon >= BNDRYE)) THEN
    IF ((ylat <= BNDRYN) AND (ylat >= BNDRYS)) THEN

        'if so, then determine the distance between the point and the
        'center of the circles.

        idir = 0
        irad = 0
        CALL GETDIS(ylat)
        CALL GETRAD(ylat, NUMRAD, irad)

        'If the point lies outside of the farthest radii, proceed to the
        'next point.

        IF (irad = 0) THEN
            GOTO Continue2
        ELSE

            'Determine which sector the point lies in.

            CALL GETDIR(ylat, idir)

            'Add the population of the block to the total population of
            'the sector in which it lies.
```

```

        sector_population(idir, irad) = sector_population(idir, irad) +
(ipop * SCALE)
        sector_area(idir, irad) = sector_area(idir, irad) + area
        sector_frcLnd(idir, irad) = sector_frcLnd(idir, irad) + (area *
county_frcLnd(county_code))

        'Increment the regional economic data by incrementing each of
        'the appropriate elements in the user defined data type.

        econ_data(region_index(irad, idir - 1)).region_area =
econ_data(region_index(irad, idir - 1)).region_area + area
        econ_data(region_index(irad, idir - 1)).region_frmfrc =
econ_data(region_index(irad, idir - 1)).region_frmfrc + (area *
county_frmfrc(county_code))
        econ_data(region_index(irad, idir - 1)).region_dpf =
econ_data(region_index(irad, idir - 1)).region_dpf + (area *
county_dpf(county_code))
        econ_data(region_index(irad, idir - 1)).region_asfp =
econ_data(region_index(irad, idir - 1)).region_asfp + (area *
county_asfp(county_code))
        econ_data(region_index(irad, idir - 1)).region_vfrm =
econ_data(region_index(irad, idir - 1)).region_vfrm + (area *
county_vfrm(county_code))
        econ_data(region_index(irad, idir - 1)).region_vnfrm =
econ_data(region_index(irad, idir - 1)).region_vnfrm + (area *
county_vnfrm(county_code))

        'Increment the number of records processed.

        census_records = census_records + 1

        IF (census_records <= 1000) THEN
            'PRINT #debug_file, USING "###.### ##.### ##### ## ##";
            ylat; ipop; idir; irad
        END IF

        IF census_records MOD 100 = 0 THEN
            'Process any user events that may have occurred while
            'calculating. At this point in time only the close
            'calculation button is allowed.

            events_open = DOEVENTS()

            'If the user has selected close from the calculate
            'form, then close the census file and exit the
            'subroutine.

            IF (leave_calculation) THEN
                CLOSE #census_file
                leave_calculation = FALSE
                EXIT SUB
            END IF

            'Update the number of census records displayed
            'status line.

            frmCalculate.lblStatus.caption =
                LTRIM$(STR$(census_records)) + -
                " census records processed."

```

```

'Read the timer.

time_end = TIMER

'Determine how many census records have been read
'(including those which were not actually
'processed).

num_processed =
    ((ibyte - start_file_pos) / record_length)

'Determine the average time per record and how long
'to process all remaining records.

total_time = (((time_end - time_begin) *
    (num_records - num_processed)) / num_processed)

'Convert time from seconds to hours/minutes/seconds.

hours = total_time \ 3600
minutes = (total_time \ 60) - (hours * 60)
seconds =
    (total_time - ((hours * 60 + minutes) * 60))

'If the time has changed by more than 5 seconds,
'then update the display.

IF (ABS(seconds - prev_seconds) > 5) THEN

    IF hours = 0 THEN
        frmCalculate.lblTime.caption =
            "Minutes:" + STR$(minutes) +
            " Seconds:" + STR$(seconds)
    ELSE
        frmCalculate.lblTime.caption =
            "Hours:" + STR$(hours) +
            " Minutes:" + STR$(minutes) +
            " Seconds:" + STR$(seconds)
    END IF

    prev_seconds = seconds

END IF

END IF

END IF

GOTO readit

ELSE

    GOTO finishup

END IF

Continue2:

GOTO readit

finishup:

```

```

'Close census data file, and update display labels.

CLOSE #census_file
'CLOSE #debug_file

frmCalculate.lblStatus.caption = "Calculating economic data."

'Determine the land fraction values for each sector.

FOR i = 1 TO number_of_segments STEP 1
    FOR j = 1 TO number_of_radii STEP 1
        IF (sector_area(i, j) <> 0) THEN
            sector_frclnd(i, j) = sector_frclnd(i, j) / sector_area(i, j)
        ELSE
            sector_frclnd(i, j) = 0
        END IF
    NEXT j
NEXT i

'Determine the economic values for each region.

FOR i = 1 TO number_econ_regions STEP 1
    IF (econ_data(i).region_area <> 0) THEN
        econ_data(i).region_frmfrc = econ_data(i).region_frmfrc /
econ_data(i).region_area
        econ_data(i).region_dpf = econ_data(i).region_dpf /
econ_data(i).region_area
        econ_data(i).region_asfp = econ_data(i).region_asfp /
econ_data(i).region_area
        econ_data(i).region_vfrm = econ_data(i).region_vfrm /
econ_data(i).region_area
        econ_data(i).region_vnfrm = econ_data(i).region_vnfrm /
econ_data(i).region_area
    ELSE
        econ_data(i).region_frmfrc = 0
        econ_data(i).region_dpf = 0
        econ_data(i).region_asfp = 0
        econ_data(i).region_vfrm = 0
        econ_data(i).region_vnfrm = 0
    END IF
NEXT i

'Update the user display for the final time letting the user know
'That we are finished and the total number of census records
'processed.

frmCalculate.lblStatus.caption = "Finished, " +
    LTRIM$(STR$(census_records)) + " processed."

```

```

'CLOSE (debug_file)

END SUB

SUB popdensity (exceeded AS INTEGER)

  'Define local variables

  DIM DELLON, DELLAT, ilon AS INTEGER, ilat AS INTEGER, i AS INTEGER
  DIM ILIN AS INTEGER, open_windows AS INTEGER
  DIM census_file AS LONG
  DIM tempstring AS STRING

  on local error goto popdensity_error

  'Initialize population array.

  for i = 1 to numrad step 1
    population(i) = 0
  next i

  'Retrieve path for census database from setup, and open database for
  'input.

  tempstring = frmSetup.txtData_Path.text + "\CENSUS90.DAT"
  OPEN tempstring FOR BINARY ACCESS READ AS #2 LEN = 32767

  DGTORD = pi / 180!

  'Set the latitude of the tropic of cancer

  AMINLA = 23.45

  'Set dismax to the farthest distance, which is the radii provided by
  'the user.

  DISMAX = RADDIS(NUMRAD)

  'Find the longitude-latitude boundaries of the grid

  CALL GETDIS(SLAT)
  SDPDLA = DPDLAT

  'Find number of degrees longitude traversed

  DELLON = DISMAX / DPDLON

  'Find the eastern and Western boundries of this section

  BNDRYW = INT((SLON + DELLON) * 1000 + .5) / 1000
  BNDRYE = INT((SLON - DELLON) * 1000 + .5) / 1000

  'Find distance change per degree latitude at some minimum latitude
  '(AMINLA).

  CALL GETDIS(AMINLA)

  'Find number of degrees latitude traversed northward

  DELLAT = DISMAX / SDPDLA

```

```

'Find northern boundry of of this section
BNDRYN = INT((SLAT + DELLAT) * 1000 + .5) / 1000

'Find the number of degrees latitude traversed southward
DELLAT = DISMAX / DPDLAT

'Find the southern boundry of this section
BNDRYS = INT((SLAT - DELLAT) * 1000 + .5) / 1000

'Find the first record in the file with the longitude within
'the longitudinal boundaries of the grid
'frmDensity.lblStatus.caption = "Searching for first census record."

'Reset all non-local variables to zero

ipop = 0
xlon = 0!
ylat = 0!
area = 0&
county_code = 0
idir = 0
irad = 0

'Call function to locate north western boundry in census file
CALL pointr(BYVAL BNDRYW, 2, ylat, ipop, area, county_code, ibyte)

'If location is not found, close the file and exit subroutine

IF (xlon = 0!) THEN
  CLOSE #2
  EXIT SUB
END IF

'If location is within the boundaries of the section, determine if
'it lies within current circle, and if so, process its population
'and goto the code which deals with all subsequent records

IF ((xlon <= BNDRYW) AND (xlon >= BNDRYE)) THEN
  IF ((ylat <= BNDRYN) AND (ylat >= BNDRYS)) THEN
    CALL GETDIS(ylat)
    CALL GETRAD(ylat, NUMRAD, irad)
    IF (irad = 0) THEN
      GOTO Continue12
    ELSE
      for i = irad to numrad step 1
        population(i) = population(i) + ipop
        if (population(i) >= population_threshold(i)) then
          exceeded = true
          close #2
          exit sub
        end if
      next i
    END IF
  END IF
END IF

'Read each subsequent record in the data file until
'first record with a longitude outside the longitudinal

```

```

'boundaries

Continuel2:

'frmDensity.lblStatus.caption = "Processing census records."

ReadIt2:

'If end of file is reached, close file and exit subroutine

IF (EOF(2)) THEN
  CLOSE #2
  EXIT SUB
END IF

'Jump seven bytes forward, to the next position in the census file

ibyte = ibyte + record_length

'Take from the current position (ibyte) in the file
'one record, which will be cut into its various pieces

GET #2, ibyte, rec

'Parse the record into longitude, latitude, population, and region

ilon = CVI(MIDS$(rec, 1, 2))
ilat = CVI(MIDS$(rec, 3, 2))
ipop = CVI(MIDS$(rec, 5, 2))
area = CVL(MIDS$(rec, 7, 4))
county_code = CVI(MIDS$(rec, 11, 2))

'Convert longitude from file format into standard decimal degrees

xlon = (ilon + longitude_offset) / 1000!

'Convert latitude from file format into standard decimal degrees

ylat = (ilat + latitude_offset) / 1000!

'If latitude and longitude are within this section, verify that
'they are actually within current circle, and if so add its
'population to the correct sector

IF ((xlon <= BNDRYW) AND (xlon >= BNDRYE)) THEN
  IF ((ylat <= BNDRYN) AND (ylat >= BNDRYS)) THEN
    idir = 0
    irad = 0
    GETDIS (ylat)
    CALL GETRAD(ylat, NUMRAD, irad)
    IF (irad = 0) THEN
      GOTO ReadIt2
    ELSE
      for i = irad to numrad step 1
        population(i) = population(i) + ipop
        if (population(i) >= population_threshold(i)) then
          exceeded = true
          close #2
          exit sub
        end if
      next i
    END IF
  END IF
END IF

```

```
'If the end of file has been reached, exit the subroutine
ELSEIF (EOF(2)) THEN
    CLOSE #2
    EXIT SUB

'If longitude is outside of the specified segment, exit the
'subroutine.

ELSE CLOSE #2
    EXIT SUB
END IF

'If longitude is still within bounds, read in the next record
GOTO ReadIt2

popdensity_error:
msgbox "Popdensity error" + chr$(13) + error$(err) + chr$(13) + _
    "At line # " + str$(erl)
    close #2

exit sub

END SUB
```

FILENAME: CALCFORM.FRM

```
Version 1.00
BEGIN Form frmCalculate
    AutoRedraw      = -1
    BackColor       = QBColor(3)
    BorderStyle     = 1
    Caption         = "Calculate"
    ControlBox      = 0
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(16)
    Left             = Char(19)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag              = ""
    Top              = Char(3)
    Visible         = -1
    Width            = Char(41)
    WindowState     = 0
    BEGIN Label Label1
        Alignment      = 0
        AutoSize       = 0
        BackColor      = QBColor(3)
        BorderStyle    = 0
        Caption         = "Calculation Status"
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(1)
        Left             = Char(1)
        MousePointer    = 0
        TabIndex        = 6
        Tag              = ""
        Top              = Char(1)
        Visible         = -1
        Width            = Char(18)
    END
    BEGIN Label Label2
        Alignment      = 0
        AutoSize       = 0
        BackColor      = QBColor(3)
        BorderStyle    = 0
        Caption         = "Estimated Time Remaining"
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(1)
        Left             = Char(1)
        MousePointer    = 0
        TabIndex        = 7
        Tag              = ""
        Top              = Char(6)
        Visible         = -1
        Width            = Char(24)
    END
    BEGIN Frame Frame1
        BackColor       = QBColor(3)
        Caption         = ""
        DragMode        = 0
        Enabled         = -1
    END
```

```

ForeColor      = QBColor(0)
Height        = Char(3)
Left          = Char(1)
MousePointer  = 0
TabIndex      = 2
Tag           = ""
Top           = Char(2)
Visible       = -1
Width         = Char(37)
BEGIN Label lblStatus
    Alignment    = 0
    AutoSize     = 0
    BackColor    = QBColor(3)
    BorderStyle  = 0
    Caption      = ""
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(1)
    Left         = Char(0)
    MousePointer = 0
    TabIndex     = 3
    Tag          = ""
    Top          = Char(0)
    Visible      = -1
    Width        = Char(35)
END
BEGIN Frame Frame2
    BackColor    = QBColor(3)
    Caption      = ""
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(3)
    Left         = Char(1)
    MousePointer = 0
    TabIndex     = 4
    Tag          = ""
    Top          = Char(7)
    Visible      = -1
    Width        = Char(37)
BEGIN Label lblTime
    Alignment    = 0
    AutoSize     = 0
    BackColor    = QBColor(3)
    BorderStyle  = 0
    Caption      = " "
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(1)
    Left         = Char(0)
    MousePointer = 0
    TabIndex     = 5
    Tag          = ""
    Top          = Char(0)
    Visible      = -1
    Width        = Char(35)
END
BEGIN CommandButton cmdCalculate
    BackColor    = QBColor(3)

```

```

        Cancel      = 0
        Caption     = "&Calculate"
        Default     = -1
        DragMode    = 0
        Enabled     = -1
        Height      = Char(3)
        Left        = Char(1)
        MousePointer = 0
        TabIndex    = 0
        TabStop     = -1
        Tag         = ""
        Top         = Char(11)
        Visible     = -1
        Width       = Char(16)
    END
    BEGIN CommandButton cmdClose
        BackColor   = QBColor(3)
        Cancel      = -1
        Caption     = "C&close"
        Default     = 0
        DragMode    = 0
        Enabled     = -1
        Height      = Char(3)
        Left        = Char(22)
        MousePointer = 0
        TabIndex    = 1
        TabStop     = -1
        Tag         = ""
        Top         = Char(11)
        Visible     = -1
        Width       = Char(16)
    END
END
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

'This form handles much of the overhead associated
'with regional calculations, including the user
'interface, reporting, pausing, and canceling

SUB cmdCalculate_Click ()
    'This subroutine drives the function that performs site calculations
    'It calls the popcalc function and keeps track of the amount of
    'time spent on a site calculation.

    CONST seconds_in_a_day = 86400

    DIM time_begin AS SINGLE, time_end AS SINGLE, total_time AS SINGLE
    DIM hours AS INTEGER, minutes AS INTEGER, seconds AS SINGLE

    'Disable calculate button on form and reset the leave_calculations
    'flag.

    cmdCalculate.Enabled = FALSE
    leave_calculation = FALSE

    'Read the timer.

    time_begin = TIMER

    'Call function to perform the actual calculation.

```

```

CALL popcalc

'If the form is still visible, ie it wasn't closed before the
'calculation was complete, then perform the following actions.

IF (frmCalculate.Visible) THEN

    'Read the timer.

    time_end = TIMER

    'Convert the time into standard format.

    IF time_end < time_begin THEN
        total_time = time_end - time_begin + seconds_in_a_day
    ELSE
        total_time = time_end - time_begin
    END IF

    hours = total_time \ 3600
    minutes = (total_time \ 60) - (hours * 60)
    seconds = (total_time - ((hours * 60 + minutes) * 60))

    'Display time to user.

    frmCalculate.lblTime.Caption = "Total processing time =" + STR$(hours) + ":" +
+ LTRIM$(STR$(minutes)) + ":" + FORMAT$(seconds, "0.00")

    'Enable calculate button.

    cmdCalculate.Enabled = TRUE

END IF

END SUB

SUB cmdClose_Click ()

    'This procedure allows the user to exit a calculation
    'before it has been completed, and it allows the
    'user to close the calculate window when a
    'calculation is finished.

    DIM reply AS INTEGER

    'If a calculation is not in progress then unload the calculate form.

    IF frmCalculate.cmdCalculate.Enabled = TRUE THEN

        UNLOAD frmCalculate

    ELSE

        'If the calculation is in progress ask the user if they really
        'want to leave.

        reply = MSGBOX("Do you really want to stop the present calculation?",

MB_YESNO + MB_DEFBUTTON1, "Close")

        IF (reply = IDYES) THEN

            UNLOAD frmCalculate

```

```
    leave_calculation = TRUE

    END IF

    END IF

END SUB

SUB Form_Load ()

    'Initialize status and time lables.

    frmCalculate.lblStatus.Caption = "Press Calculate to start."
    frmCalculate.lblTime.Caption = ""

    'Enable the Calculate button and reset the leave calculation flag.

    frmCalculate.cmdCalculate.Enabled = TRUE
    leave_calculation = FALSE

END SUB
```

**FILENAME: CIRCLE.FRM**

```
Version 1.00
BEGIN Form frmMake_a_Circle
    AutoRedraw      = 0
    BackColor       = QBColor(3)
    BorderStyle     = 1
    Caption         = "Regional Population Densities Above The Threshold"
    ControlBox      = 0
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(21)
    Left            = Char(9)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag             = ""
    Top             = Char(1)
    Visible         = -1
    Width           = Char(59)
    WindowState     = 0
    BEGIN Label Label7
        Alignment      = 0
        AutoSize       = -1
        BackColor      = QBColor(3)
        BorderStyle    = 0
        Caption         = "Display results as:"
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(1)
        Left            = Char(2)
        MousePointer   = 0
        TabIndex        = 13
        Tag             = ""
        Top             = Char(14)
        Visible         = -1
        Width           = Char(19)
    END
    BEGIN TextBox txtSpacing
        BackColor      = QBColor(3)
        BorderStyle    = 1
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(3)
        Left            = Char(26)
        MousePointer   = 0
        MultiLine       = 0
        ScrollBars      = 0
        TabIndex        = 3
        TabStop         = -1
        Tag             = ""
        Text            = ""
        Top             = Char(3)
        Visible         = -1
        Width           = Char(21)
    END
    BEGIN TextBox txtThreshold
        BackColor      = QBColor(3)
        BorderStyle    = 1
        DragMode        = 0
    END
```

```

Enabled      = -1
ForeColor    = QBColor(0)
Height       = Char(3)
Left         = Char(2)
MousePointer = 0
MultiLine    = 0
ScrollBars   = 0
TabIndex     = 5
TabStop      = -1
Tag          = ""
Text         = ""
Top          = Char(10)
Visible      = -1
Width        = Char(21)

END
BEGIN Label Label2
    Alignment   = 0
    AutoSize    = 0
    BackColor   = QBColor(3)
    BorderStyle = 0
    Caption     = "&Input Map Name"
    DragMode    = 0
    Enabled     = -1
    ForeColor   = QBColor(0)
    Height      = Char(1)
    Left         = Char(29)
    MousePointer = 0
    TabIndex    = 6
    Tag          = ""
    Top          = Char(9)
    Visible     = -1
    Width        = Char(14)

END
BEGIN TextBox txtInput_Map
    BackColor   = QBColor(3)
    BorderStyle = 1
    DragMode    = 0
    Enabled     = -1
    ForeColor   = QBColor(0)
    Height      = Char(3)
    Left         = Char(26)
    MousePointer = 0
    MultiLine   = 0
    ScrollBars  = 0
    TabIndex    = 7
    TabStop     = -1
    Tag          = ""
    Text         = ""
    Top          = Char(10)
    Visible     = -1
    Width        = Char(21)

END
BEGIN CommandButton cmdInput_files
    BackColor   = QBColor(3)
    Cancel      = 0
    Caption     = "&Files"
    Default     = 0
    DragMode    = 0
    Enabled     = -1
    Height      = Char(3)
    Left         = Char(48)
    MousePointer = 0
    TabIndex    = 8

```

```

        TabStop      = -1
        Tag          = ""
        Top          = Char(10)
        Visible      = -1
        Width         = Char(7)

    END
    BEGIN OptionButton optPoints
        BackColor    = QBColor(3)
        Caption      = "&Points"
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height        = Char(1)
        Left          = Char(36)
        MousePointer = 0
        TabIndex      = 10
        TabStop       = 0
        Tag           = ""
        Top           = Char(14)
        Value          = 0
        Visible       = -1
        Width         = Char(10)

    END
    BEGIN OptionButton optCircles
        BackColor    = QBColor(3)
        Caption      = "&Circles"
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height        = Char(1)
        Left          = Char(23)
        MousePointer = 0
        TabIndex      = 9
        TabStop       = -1
        Tag           = ""
        Top           = Char(14)
        Value          = -1
        Visible       = -1
        Width         = Char(12)

    END
    BEGIN CommandButton cmdCalculate_Regional
        BackColor    = QBColor(3)
        Cancel       = 0
        Caption      = "C&alculate"
        Default      = 0
        DragMode     = 0
        Enabled      = -1
        Height        = Char(3)
        Left          = Char(10)
        MousePointer = 0
        TabIndex      = 11
        TabStop       = -1
        Tag           = ""
        Top           = Char(16)
        Visible       = -1
        Width         = Char(17)

    END
    BEGIN CommandButton cmdClose
        BackColor    = QBColor(3)
        Cancel       = -1
        Caption      = "C&close"
        Default      = 0
        DragMode     = 0

```

```

Enabled      = -1
Height       = Char(3)
Left         = Char(30)
MousePointer = 0
TabIndex     = 12
TabStop      = -1
Tag          = ""
Top          = Char(16)
Visible      = -1
Width        = Char(17)
END
BEGIN TextBox txtRadii
BackColor    = QBColor(3)
BorderStyle  = 1
DragMode     = 0
Enabled      = -1
ForeColor    = QBColor(0)
Height       = Char(3)
Left         = Char(2)
MousePointer = 0
MultiLine    = 0
ScrollBars   = 0
TabIndex     = 1
TabStop      = -1
Tag          = ""
Text          = ""
Top          = Char(3)
Visible      = -1
Width        = Char(21)
END
BEGIN Label Label1
Alignment    = 0
AutoSize     = 0
BackColor    = QBColor(3)
BorderStyle  = 0
Caption      = "&Radii of Circles (mi)"
DragMode     = 0
Enabled      = -1
ForeColor    = QBColor(0)
Height       = Char(1)
Left         = Char(2)
MousePointer = 0
TabIndex     = 0
Tag          = ""
Top          = Char(2)
Visible      = -1
Width        = Char(21)
END
BEGIN Label Label3
Alignment    = 2
AutoSize     = 0
BackColor    = QBColor(3)
BorderStyle  = 0
Caption      = "Longitudinal &Spacing of Circles (mi)"
DragMode     = 0
Enabled      = -1
ForeColor    = QBColor(0)
Height       = Char(2)
Left         = Char(26)
MousePointer = 0
TabIndex     = 2
Tag          = ""
Top          = Char(1)

```

```

        Visible      = -1
        Width       = Char(20)
    END
    BEGIN Label Label4
        Alignment    = 2
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 0
        Caption      = "Population &Density Threshold (People/Square mi)"
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Left         = Char(3)
        MousePointer = 0
        TabIndex     = 4
        Tag          = ""
        Top          = Char(7)
        Visible      = -1
        Width        = Char(18)
    END
END
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

'This form deals with much of the overhead associated with regional
'calculations, including the user interface parsing the data into a
'usable format, and reporting the status of the calculation.

SUB cmdCalculate_Regional_Click ()
    'Declare a few local variables. Most of these variables are required
    'to put the data in a format acceptable to mapplan.

    DIM tempstring AS STRING, tempchar AS STRING
    DIM orig_input AS STRING
    DIM number_of_circle_segments AS INTEGER, i AS INTEGER
    DIM x_int AS INTEGER, y_int AS INTEGER, spacing AS DOUBLE
    DIM rotation_increment AS DOUBLE, radii AS DOUBLE, x AS DOUBLE
    DIM y_radius AS DOUBLE, y_space AS DOUBLE
    DIM y0 AS DOUBLE, y2 AS DOUBLE
    DIM x_radius AS DOUBLE, x_space AS DOUBLE
    DIM x0 AS DOUBLE, x1 AS DOUBLE, x2 AS DOUBLE
    DIM xmin AS INTEGER, ymin AS INTEGER
    DIM xmax AS INTEGER, ymax AS INTEGER
    DIM two AS INTEGER, max_points AS INTEGER
    DIM last_x AS INTEGER, last_y AS INTEGER
    DIM map_name AS STRING, link_field AS STRING, offset AS INTEGER
    DIM input_map AS STRING, output_map AS STRING
    DIM output_database AS STRING
    DIM min_longitude AS DOUBLE, min_latitude AS DOUBLE
    DIM max_longitude AS DOUBLE, max_latitude AS DOUBLE
    DIM longitude_factor AS DOUBLE, latitude_factor AS DOUBLE
    DIM number_of_layers AS INTEGER
    DIM number_of_map_records AS LONG, number_of_data_records AS LONG
    DIM bytes_in_map_header AS INTEGER, bytes_in_data_header AS INTEGER
    DIM bytes_in_map_record AS INTEGER, bytes_in_data_record AS INTEGER
    DIM mid1 AS STRING * 18, layerno AS STRING * 3
    DIM object_type AS STRING * 1
    DIM stp_x AS STRING * 5, stp_y AS STRING * 5, no_elm AS STRING * 3
    DIM pos_off AS STRING * 8, nop AS STRING * 5
    DIM object_xmin AS STRING * 5

```

```

DIM object_ymin AS STRING * 5, object_xmax AS STRING * 5
DIM object_ymax AS STRING * 5, deleted AS STRING * 1
DIM mid2 AS STRING * 8, density AS STRING * 1
DIM vector_offset AS LONG
DIM threshold AS SINGLE
DIM exceeded AS INTEGER
DIM total_circles AS LONG, number_of_circles AS LONG
DIM rows AS LONG, row0 AS LONG, row1 AS LONG
DIM m AS DOUBLE, previous AS DOUBLE
DIM time_begin AS SINGLE, time_end AS SINGLE, total_time AS SINGLE
DIM hours AS INTEGER, minutes AS INTEGER, seconds AS SINGLE
DIM reply as integer

'Check for valid input from the user.

SELECT CASE verify_input(3)
  CASE 0
    frmMake_a_circle.txtRadii.SETFOCUS
    EXIT SUB
  CASE -1
    frmMake_a_circle.txtspacing.SETFOCUS
    EXIT SUB
  CASE -2
    frmMake_a_circle.txtThreshold.SETFOCUS
    EXIT SUB
  CASE -3
    frmMake_a_circle.txtinput_map.SETFOCUS
    EXIT SUB
END SELECT

'Start timer.

time_begin = TIMER

'Put filenames into an acceptable format.
'Remove extension from input map name, if necessary. This section
'allows the user to enter file names on frmMake_a_circle with or
'without extension. If an extension is present, it is temporarily
'removed from the file name, so that secpop can append the necessary
'extensions for the various map files. The filename and extension
'offered by the user are preserved.

tempstring = ""
FOR i = 1 TO (LEN(frmMake_a_circle.txtinput_map.text))
  tempchar = MID$(frmMake_a_circle.txtinput_map.text, i, 1)
  IF tempchar = "." THEN
    EXIT FOR
  ELSE
    tempstring = tempstring + tempchar
  END IF
NEXT i
orig_input = frmMake_a_circle.txtinput_map.text
frmMake_a_circle.txtinput_map.text = tempstring

'Set up final error checking to catch file names that are too long
'and anything else that may have slipped by.

ON LOCAL ERROR GOTO regional_file_error

'Read in the user input. Convert miles to kilometers.

spacing = VAL(frmMake_a_circle.txtspacing.text) * _
  miles_to_kilometers

```

```

threshold = VAL(frmMake_a_circle.txtThreshold.text) / _
(miles_to_kilometers ^ 2)

'Set the radii to be equal to 1 mile increments out to 20 miles
'and 10 mile increments after that. 170 miles is the largest
'possible radii. Convert the miles to kilometers.

radii = val(txtRadii.text)
for i = 1 to max_number_of_radii step 1
    if (regional_radii(i) < radii) then
        raddis(i) = regional_radii(i) * miles_to_kilometers
    else
        raddis(i) = radii * miles_to_kilometers
    numrad = i
    exit for
end if
next i
radii = radii * miles_to_kilometers

'Initialize population threshold array.

for i = 1 to numrad step 1
    population_threshold(i) = fix(threshold * pi * _
        (raddis(i) ^ 2)) + 1
next i

input_map = frmMake_a_circle.txtinput_map.text
output_map = "mapplan\map_out"
output_database = "mapplan\db_out"

'Copy the inputmap .dbf and .vtr files to the output map files,
'and suppress the dos messages by dumping them into a temp file.

call copy_file(input_map + ".dbf", output_map + ".dbf")
call copy_file(input_map + ".spc", output_map + ".spc")
call copy_file(input_map + ".vtr", output_map + ".vtr")
call copy_file("mapplan\database.hdr", output_database + ".dbf")

'Open the input map .cfg file and read in the data from it.

OPEN input_map + ".cfg"
    FOR INPUT ACCESS READ LOCK READ WRITE AS #30 LEN = 1024
INPUT #30, xmin, ymin, xmax, ymax, two, max_points
LINE INPUT #30, map_name
LINE INPUT #30, link_field
INPUT #30, min_longitude, min_latitude, max_longitude, max_latitude
INPUT #30, longitude_factor, latitude_factor
CLOSE #30

'Open the input map .lyr file, and read in the data from it.

OPEN input_map + ".lyr"
    FOR INPUT ACCESS READ LOCK READ WRITE AS #30 LEN = 1024
i = 1

'Read in information on each layer of the map until end of file is
'reached.

WHILE (NOT EOF(30))
    i = i + 1
    INPUT #30, layer_number(i), data_flag(i), display_flag(i)
    INPUT #30, active_layer(i), layer_name(i)
    INPUT #30, pts_color(i), pts_type(i), pts_size(i), pts_mode(i)

```

```

    INPUT #30, lines_color(i), lines_type(i), lines_size(i), _
    lines_mode(i)
    INPUT #30, polyg_color(i), polyg_type(i), polyg_size(i), _
    polyg_mode(i)
    INPUT #30, text_color(i), text_type(i), text_size(i), _
    text_mode(i)
    layer_number(i) = layer_number(i) + 1
    data_flag(i) = 0
    active_layer(i) = 0
WEND
number_of_layers = i
CLOSE #30

'Open .lyr output map, put values to put in this file into
'a set of variables, and write these variables to the file.

OPEN output_map + ".lyr"
    FOR OUTPUT ACCESS WRITE LOCK READ WRITE AS #30 LEN = 1024

layer_number(1) = 1
data_flag(1) = 2
display_flag(1) = 1
active_layer(1) = 1
layer_name(1) = "CIRCLES"
pts_color(1) = 6
pts_type(1) = 0
pts_size(1) = 1
pts_mode(1) = 0
lines_color(1) = 6
lines_type(1) = 1
lines_size(1) = 1
lines_mode(1) = 0
polyg_color(1) = 6
polyg_type(1) = 1
polyg_size(1) = 1
polyg_mode(1) = 0
text_color(1) = 0
text_type(1) = 0
text_size(1) = 1
text_mode(1) = 0

'Print first layer data to file.

PRINT #30, MID$(STR$(layer_number(1)), 2); ",";
PRINT #30, MID$(STR$(data_flag(1)), 2); ",";
PRINT #30, MID$(STR$(display_flag(1)), 2); ",";
PRINT #30, MID$(STR$(active_layer(1)), 2); ",";
PRINT #30, layer_name(1); ",";
PRINT #30, MID$(STR$(pts_color(1)), 2); ",";
PRINT #30, MID$(STR$(pts_type(1)), 2); ",";
PRINT #30, MID$(STR$(pts_size(1)), 2); ",";
PRINT #30, MID$(STR$(pts_mode(1)), 2); ",";
PRINT #30, MID$(STR$(lines_color(1)), 2); ",";
PRINT #30, MID$(STR$(lines_type(1)), 2); ",";
PRINT #30, MID$(STR$(lines_size(1)), 2); ",";
PRINT #30, MID$(STR$(lines_mode(1)), 2); ",";
PRINT #30, MID$(STR$(polyg_color(1)), 2); ",";
PRINT #30, MID$(STR$(polyg_type(1)), 2); ",";
PRINT #30, MID$(STR$(polyg_size(1)), 2); ",";
PRINT #30, MID$(STR$(polyg_mode(1)), 2); ",";
PRINT #30, MID$(STR$(text_color(1)), 2); ",";
PRINT #30, MID$(STR$(text_type(1)), 2); ",";
PRINT #30, MID$(STR$(text_size(1)), 2); ","

```

```

PRINT #30, MID$(STR$(text_mode(1)), 2)

'Print all other layers to file.

FOR i = 2 TO number_of_layers STEP 1
    PRINT #30, MID$(STR$(layer_number(i)), 2); ",";
    PRINT #30, MID$(STR$(data_flag(i)), 2); ",";
    PRINT #30, MID$(STR$(display_flag(i)), 2); ",";
    PRINT #30, MID$(STR$(active_layer(i)), 2); ",";
    PRINT #30, layer_name(i); ",";
    PRINT #30, MID$(STR$(pts_color(i)), 2); ",";
    PRINT #30, MID$(STR$(pts_type(i)), 2); ",";
    PRINT #30, MID$(STR$(pts_size(i)), 2); ",";
    PRINT #30, MID$(STR$(pts_mode(i)), 2); ",";
    PRINT #30, MID$(STR$(lines_color(i)), 2); ",";
    PRINT #30, MID$(STR$(lines_type(i)), 2); ",";
    PRINT #30, MID$(STR$(lines_size(i)), 2); ",";
    PRINT #30, MID$(STR$(lines_mode(i)), 2); ",";
    PRINT #30, MID$(STR$(polyg_color(i)), 2); ",";
    PRINT #30, MID$(STR$(polyg_type(i)), 2); ",";
    PRINT #30, MID$(STR$(polyg_size(i)), 2); ",";
    PRINT #30, MID$(STR$(polyg_mode(i)), 2); ",";
    PRINT #30, MID$(STR$(text_color(i)), 2); ",";
    PRINT #30, MID$(STR$(text_type(i)), 2); ",";
    PRINT #30, MID$(STR$(text_size(i)), 2); ",";
    PRINT #30, MID$(STR$(text_mode(i)), 2)

NEXT i
CLOSE #30

'Place degrees to radians conversion factor into DGTORD (DeGrees TO
'RaDians).

DGTORD = pi / 180!

'Set Distance per degree longitude/latitude at average latitude.

CALL GETDIS(CSNG((min_latitude + max_latitude) / 2#))

'Set up variables used to determine how far apart vertically (rows,
'row*) the circles are, and how far apart horizontally (*_radius,
'*_space) these same circles are.

x_radius = ((radii / DPDLON) / longitude_factor) * 1000000#
y_radius = ((radii / DPDLAT) / latitude_factor) * 1000000#
x_space = ((spacing / DPDLON) / longitude_factor) * 1000000#
y_space = ((spacing / DPDLAT) / latitude_factor) * 1000000# / 2#
rows = -INT(-(ymax + y_radius - ymin + 1!) / y_space)
row0 = -INT(-(xmax + x_radius - xmin + 1!) / x_space)
row1 = -INT(-(xmax + x_radius - xmin - (x_space / 2!) + 1!) / -
    x_space)

'Determine total number of circles.

total_circles =
    INT((rows / 2!) + .75) * row0 + INT((rows / 2!)) * row1

'Open all output map files.

OPEN output_map + ".dbf"
    FOR BINARY ACCESS READ WRITE LOCK READ WRITE AS #30
OPEN output_map + ".vtr"
    FOR BINARY ACCESS READ WRITE LOCK READ WRITE AS #31

```

```

OPEN output_database + ".dbf"
    FOR BINARY ACCESS READ WRITE LOCK READ WRITE AS #32
OPEN output_map + ".cfg"
    FOR OUTPUT ACCESS WRITE LOCK READ WRITE AS #33 LEN = 1024

'Read in and process various bits of data from files.

GET #30, 5&, number_of_map_records
GET #30, , bytes_in_map_header
GET #30, , bytes_in_map_record

GET #32, 5&, number_of_data_records
GET #32, , bytes_in_data_header
GET #32, , bytes_in_data_record

'Initialize file related variables.

vector_offset = LOF(31)
deleted = " "
layerno = " 1"
no Elm = " 1"
object_type = "3"

IF (frmMake_a_circle.optCircles.value) THEN

    'Determine the number of "sides" on each circle.

    x = 1# / (4# * radii)
    number_of_circle_segments =
        INT(pi / ATN(x / SQR(1# - x * x)) + 1#)
    IF (number_of_circle_segments < 10) THEN
        number_of_circle_segments = 10
    END IF

    'Determine the number of vertices for each circle.

    nop = RIGHT$("      " + STR$(number_of_circle_segments + 1#), 5)

    'Determine the rotational increment for each circle.

    rotation_increment = 2# * pi / number_of_circle_segments

    'Update the maximum number of vertices for the map if necessary.

    IF (max_points < number_of_circle_segments) THEN
        max_points = number_of_circle_segments + 1
    END IF

    ELSE

        'If the user has selected "points" from the circle form, set
        'number of vertices to 3, and ignore all of the circle
        'variables. This will give the user a triangle, which is the
        'smallest item that mapplan can process.

        nop = "3"

    END IF

    'Print data to the .cfg map file.

    PRINT #33, MID$(STR$(xmin), 2); ymin; xmax; ymax; two; max_points

```

```

PRINT #33, map_name
PRINT #33, link_field
PRINT #33, min_longitude; min_latitude; max_longitude; max_latitude;
PRINT #33, longitude_factor; latitude_factor
CLOSE #33

'Print data to all of the other map files.

density = "1"

SEEK #30, bytes_in_map_header + 1#
SEEK #31, LOF(31) + 1#
SEEK #32, LOF(32)

'Initialize variables for calculation, hide user input form, and
'show density form.

y0 = ymin
last_y = FALSE
offset = 0
number_of_circles = 0
frmMake_a_circle.HIDE
frmDensity.SHOW MODELESS
frmDensity.lblPercent_complete.caption = "0%"

'While latitude and longitude are within bounds, process census data
'files.

WHILE (y0 - y_radius < ymax)
  IF (offset = 0) THEN
    x0 = xmin
    offset = 1
  ELSE
    x0 = xmin + (.5 * x_space)
    offset = 0
  END IF
  x1 = x0 + x_radius
  WHILE (x0 - x_radius < xmax)

    'Check to see if density form is still visible.  If not,
    'reset all variables, close all open files, and exit
    'subroutine.

    reply = doevents()
    IF frmDensity.visible = 0 THEN
      CLOSE
      UNLOAD frmMake_a_circle
      EXIT SUB
    END IF

    'Set starting latitude and longitude.

    SLAT = (y0 * .000001 * latitude_factor) + min_latitude
    SLON = -((x0 * .000001 * longitude_factor) + min_longitude)

    'Reset variable which tells if the calculations on a circle
    'are done or not.

    exceeded = FALSE

    'Call function to perform actual calculations.

    CALL popdensity(exceeded)

```

```

'If the number of people in the circle are greater than the
'threshold, then plot the circle in the map file, in a
'manner understood by mapplan, and increment counters.

IF (exceeded) THEN
    number_of_map_records = number_of_map_records + 1
    number_of_data_records = number_of_data_records + 1

    mid1 = LEFT$("POP" +
        MID$(STR$(number_of_data_records), 2) + _
        " ", 18)
    stp_x = RIGHT$(" " + STR$(INT(x0 + .5#)), 5)
    stp_y = RIGHT$(" " + STR$(INT(y0 + .5#)), 5)
    pos_off = RIGHT$(" " + STR$(vector_offset), 8)

    IF (INT(x0 - x_radius + .5#) >= xmin) THEN
        object_xmin = RIGHT$(" " +
            STR$(INT(x0 - x_radius + .5#)), 5)
    ELSE
        object_xmin = RIGHT$(" " + STR$(xmin), 5)
    END IF
    IF (INT(y0 - y_radius + .5#) >= ymin) THEN
        object_ymin = RIGHT$(" " +
            STR$(INT(y0 - y_radius + .5#)), 5)
    ELSE
        object_ymin = RIGHT$(" " + STR$(ymin), 5)
    END IF
    IF (INT(x0 + x_radius + .5#) <= xmax) THEN
        object_xmax = RIGHT$(" " +
            STR$(INT(x0 + x_radius + .5#)), 5)
    ELSE
        object_xmax = RIGHT$(" " + STR$(xmax), 5)
    END IF
    IF (INT(y0 + y_radius + .5#) <= ymax) THEN
        object_ymax = RIGHT$(" " +
            STR$(INT(y0 + y_radius + .5#)), 5)
    ELSE
        object_ymax = RIGHT$(" " + STR$(ymax), 5)
    END IF

    'Write data to output file.

    PUT #30, , deleted
    PUT #30, , mid1
    PUT #30, , layerno
    PUT #30, , object_type
    PUT #30, , stp_x
    PUT #30, , stp_y
    PUT #30, , no_elm
    PUT #30, , pos_off
    PUT #30, , nop
    PUT #30, , object_xmin
    PUT #30, , object_ymin
    PUT #30, , object_xmax
    PUT #30, , object_ymax

    mid2 = LEFT$(mid1, 8)

    PUT #32, , deleted
    PUT #32, , mid2
    PUT #32, , density

```

```

'Plot the actual points for the circle, and place them
'into the vector file.

IF (frmMake_a_circle.optCircles.value) THEN
    x_int = INT(x1 + .5#)
    y_int = INT(y0 + .5#)
    PUT #31, , x_int
    PUT #31, , y_int
    vector_offset = vector_offset + 4#

    FOR i = 1 TO number_of_circle_segments STEP 1
        x2 = x_radius * COS(i * rotation_increment) + x0
        y2 = y_radius * SIN(i * rotation_increment) + y0
        IF (x2 <= xmin) THEN x_int = INT(xmin + .1)
        IF (x2 >= xmax) THEN y_int = INT(ymin + .1)
        IF (y2 <= ymin) THEN x_int = INT(xmin - .1)
        IF (y2 >= ymax) THEN y_int = INT(ymin - .1)
        x_int = INT(x2 + .5#)
        y_int = INT(y2 + .5#)
        vector_offset = vector_offset + 4#
        PUT #31, , x_int
        PUT #31, , y_int
    NEXT i
ELSE
    'Draw very small triangles instead of circles if
    'that is what the user has requested. This method
    'is much faster than plotting circles.

    x2 = (x0 - 5#)
    y2 = (y0 - 4.33#)
    IF (x2 <= xmin) THEN x_int = INT(xmin + .1)
    IF (x2 >= xmax) THEN y_int = INT(ymin + .1)
    IF (y2 <= ymin) THEN x_int = INT(xmin - .1)
    IF (y2 >= ymax) THEN y_int = INT(ymin - .1)
    x_int = INT(x2 + .5#)
    y_int = INT(y2 + .5#)
    PUT #31, , x_int
    PUT #31, , y_int

    y2 = (y0 + 4.33#)
    IF (x2 >= xmax) THEN y_int = INT(ymin + .1)
    IF (y2 >= ymax) THEN y_int = INT(ymin - .1)
    y_int = INT(y2 + .5#)
    x_int = INT(x0 + .5#)
    PUT #31, , x_int
    PUT #31, , y_int

    x2 = (x0 + 5#)
    y2 = (y0 - 4.33#)
    IF (x2 <= xmin) THEN x_int = INT(xmin + .1)
    IF (x2 >= xmax) THEN y_int = INT(ymin + .1)
    IF (y2 <= ymin) THEN x_int = INT(xmin - .1)
    IF (y2 >= ymax) THEN y_int = INT(ymin - .1)
    x_int = INT(x2 + .5#)
    y_int = INT(y2 + .5#)
    PUT #31, , x_int
    PUT #31, , y_int

    vector_offset = vector_offset + 12#
END IF
END IF

```

```

'Increment the number of circles, and tell the user what %
'of calc is complete.

    number_of_circles = number_of_circles + 1
    IF ((number_of_circles / total_circles * 100) < .1) THEN
        frmDensity.lblPercent_complete.caption = "0 %"
    ELSE
        frmDensity.lblPercent_complete.caption = _
            LEFT$(STR$(number_of_circles / _
                total_circles * 100), 5) + "%"
    x0 = x0 + x_space
    x1 = x0 + x_radius
    END IF
    WEND
    y0 = y0 + y_space
WEND

'When this point is reached, the calculation is finished. This part
'finishes up writing to the files, and closes all files.

PUT #32, 5&, number_of_data_records
CLOSE 31
CLOSE 32

OPEN input_map + ".dbf"
    FOR BINARY ACCESS READ WRITE LOCK READ WRITE AS #31
GET #31, 9&, bytes_in_map_header
SEEK #31, bytes_in_map_header + 1#

WHILE (NOT EOF(31))
    GET #31, , deleted
    GET #31, , mid1
    GET #31, , layerno
    layerno = RIGHT$(" " + STR$(VAL(layerno) + 1), 3)
    GET #31, , object_type
    GET #31, , stp_x
    GET #31, , stp_y
    GET #31, , no_elm
    GET #31, , pos_off
    GET #31, , nop
    GET #31, , object_xmin
    GET #31, , object_ymin
    GET #31, , object_xmax
    GET #31, , object_ymax

    PUT #30, , deleted
    PUT #30, , mid1
    PUT #30, , layerno
    PUT #30, , object_type
    PUT #30, , stp_x
    PUT #30, , stp_y
    PUT #30, , no_elm
    PUT #30, , pos_off
    PUT #30, , nop
    PUT #30, , object_xmin
    PUT #30, , object_ymin
    PUT #30, , object_xmax
    PUT #30, , object_ymax
WEND

PUT #30, 5&, number_of_map_records
CLOSE 30
CLOSE 31

```

```

'Stop timer.

time_end = TIMER

'Convert the time into standard format.

IF time_end <= time_begin THEN
    total_time = time_end - time_begin + seconds_in_a_day
ELSE
    total_time = time_end - time_begin
END IF
hours = total_time \ 3600
minutes = (total_time \ 60) - (hours * 60)
seconds = (total_time - ((hours * 60 + minutes) * 60))

'Display time to user.

frmDensity.cmdClose.caption = "Continue"
frmDensity.lblCaption.caption = "Total Time"
frmDensity.lblPercent_complete.caption = STR$(hours) + ":" + _
    LTRIM$(STR$(minutes)) + ":" + FORMAT$(seconds, "0.00")
WHILE (frmDensity.visible <> 0)
    reply = doevents()
WEND

'Run mapplan.

RUN "run_mppl.exe"

EXIT SUB

regional_file_error:

if(err = 6) then
    msgbox error$(err) + chr$(13) + chr$(10) +
        "In Regional Calculation Routine." + chr$(13) +
        chr$(10) + "Use smaller circles, spacing, or map."
else
    msgbox error$(err) + chr$(13) + chr$(10) +
        "In Regional Calculation Routine."
end if

'msgbox "Line number =" + str$(erl)

frmMake_a_circle.txtinput_map.text = orig_input
IF frmDensity.visible = TRUE THEN
    UNLOAD frmDensity
END IF
EXIT SUB
END SUB

SUB cmdClose_Click ()

'Close regional calculation before the calculation
'has been run.

UNLOAD frmMake_a_circle

END SUB

```

```

SUB cmdInput_files_Click ()
'This routine allows the user to select an input map file for regional
'calculations.

    DIM filenum AS INTEGER, forecolor AS INTEGER, backcolor AS INTEGER
    DIM Flags AS INTEGER, Cancel AS INTEGER
    DIM filename AS STRING
    DIM pathname AS STRING
    DIM DefaultExt AS STRING, DialogTitle AS STRING

    'Initialize file dialogue box.

    DefaultExt = "*.CFG"
    DialogTitle = "Input Map File"
    backcolor = WHITE
    forecolor = BLACK
    pathname = "mapplan\maps"

    'Prompt user for file name (from dialogue box).

    CALL FileOpen(filename, pathname, DefaultExt, DialogTitle, _
                  forecolor, backcolor, Flags, Cancel)

    'If user did not choose cancel in dialogue box...

    IF NOT Cancel THEN

        'If filename is not in current directory, append path name
        'to it.

        IF pathname <> "" THEN pathname = pathname + "\"
        filename = pathname + filename
        frmMake_a_circle.txtinput_map.text = filename

        'Set focus on next field in regional form.

        frmMake_a_circle.cmdCalculate_regional.SETFOCUS

    ELSE
        frmMake_a_circle.txtinput_map.SETFOCUS
    END IF

END SUB

SUB optCircles_Click ()
    frmMake_a_circle.optCircles.value = TRUE
    frmMake_a_circle.cmdCalculate_regional.SETFOCUS
END SUB

SUB optPoints_Click ()
    frmMake_a_circle.optPoints.value = TRUE
    frmMake_a_circle.cmdCalculate_regional.SETFOCUS
END SUB

```

FILENAME: CMNDLG.BAS

OPTION EXPLICIT

'\$INCLUDE: 'secpop90.bi'

' -----  
' Visual Basic for MS-DOS Common Dialog Toolkit

' The Common Dialog Toolkit (CMNDLG.BAS and CMNDLGF.FRM)  
' provides support for the following dialogs:

'      FileOpen  
'      FileSave  
'      FilePrint

' Support for each dialog is provided via procedures with  
' these same names that create the corresponding dialog  
' and return user input to your program. These procedures  
' only provide the user interface and return user input.  
' They do not actually carry out the corresponding actions  
' such as opening the file. Detailed descriptions of  
' these procedures are contained in the comment headers  
' above each.

' All common dialogs are created from the same form (CMNDLGF.FRM).  
' The necessary controls for each dialog are children of  
' a container picture box for the dialog. Thus the  
' form (CMNDLGF.FRM) contains a picture box with  
' appropriate controls for common dialog listed above.  
' When a particular common dialog is created and displayed,  
' the container picture box for that dialog is made visible  
' (thus all controls on that picture box become visible)  
' and the form is centered and sized to match the  
' container picture box.

' To use these common dialogs in your programs, include  
' CMNDLG.BAS and CMNDLGF.FRM in your program or use the  
' supplied library (CMNDLG.LIB, CMNDLGA.LIB - AltMath version  
' for Professional Edition only) and Quick library (CMNDLG.QLB)  
' and call the appropriate procedure to invoke the dialog  
' you need.

' Copyright (C) 1982-1992 Microsoft Corporation

' You have a royalty-free right to use, modify, reproduce  
' and distribute the sample applications and toolkits provided with  
' Visual Basic for MS-DOS (and/or any modified version)  
' in any way you find useful, provided that you agree that  
' Microsoft has no warranty, obligations or liability for  
' any of the sample applications or toolkits.

' -----  
' FileOpen common dialog support routine

' Displays Open dialog which allows users to select a  
' file from disk. This procedure only provides  
' the user interface and returns user input. It does  
' not actually carry out the corresponding action.

' Parameters:

'      FileName - returns the name (without path) of the  
'            file the user wants to open. To supply

```

' default filename in dialog, assign default
' to FileName then pass it to this procedure.
' PathName - returns the path (without filename) of
' the file the user wants to open. To supply
' default path in dialog, assign default to
' PathName then pass it to this procedure.
' Note, only pass a valid drive and path. Do
' not include a filename or file pattern.
' DefaultExt - sets the default search pattern for the
' File Listbox. Default pattern when DefaultExt
' is null is "*.*". To specify a different
' search pattern (i.e. "*.BAS"), assign new
' value to DefaultExt then pass it to this
' procedure.
' DialogTitle - sets the dialog title. Default title
' when DialogTitle is null is "Open". To
' specify a different title (i.e. "Open My File"),
' assign new value to DialogTitle then pass it to
' this procedure.
' ForeColor - sets the dialog foreground color. Does not affect
' SCREEN.ControlPanel color settings.
' BackColor - sets the dialog background color. Does not affect
' SCREEN.ControlPanel color settings.
' Flags - unused. Use this to customize dialog action if needed.
' Cancel - returns whether or not user pressed the dialog's Cancel
' button. True (-1) means the user cancelled the dialog.

SUB FileOpen (FileName AS STRING, PathName AS STRING,
             DefaultExt AS STRING, DialogTitle AS STRING, ForeColor AS INTEGER, _
             BackColor AS INTEGER, Flags AS INTEGER, Cancel AS INTEGER)

    ' Set up error handling for option validation.
    ON LOCAL ERROR GOTO FileOpenError

    ' Set form caption.
    IF DialogTitle = "" THEN
        frmCmnDlg.Caption = "Open"
    ELSE
        frmCmnDlg.Caption = DialogTitle
    END IF

    ' Determine search pattern for file listbox.
    IF DefaultExt <> "" THEN
        frmCmnDlg.filOpenList.Pattern = DefaultExt
    ELSE
        frmCmnDlg.filOpenList.Pattern = "*.*"
    END IF

    ' Determine default path.
    IF PathName <> "" THEN
        ' Set drive and path for file-system controls.
        ' Set Directory listbox path. If PathName is different
        ' than current path, PathChange event will be triggered
        ' which updates Drive listbox drive and File listbox path.
        frmCmnDlg.dirOpenList.Path = PathName
    END IF
    ' Display current path to the user.
    frmCmnDlg.lblOpenPath.Caption = frmCmnDlg.filOpenList.Path

    ' Determine default filename to display in edit field.
    IF FileName <> "" THEN
        frmCmnDlg.txtOpenFile.Text = UCASE$(FileName)
    ELSE

```

```

        frmCmnDlg.txtOpenFile.Text = frmCmnDlg.filOpenList.Pattern
END IF

' Set default and cancel command buttons.
frmCmnDlg.cmdOpenOK.Default = TRUE
frmCmnDlg.cmdOpenCancel.Cancel = TRUE

' Size and position Open/Save container.
frmCmnDlg.pctFileOpen.BorderStyle = 0
frmCmnDlg.pctFileOpen.Visible = TRUE

' Size and center dialog.
frmCmnDlg.MOVE frmCmnDlg.Left, frmCmnDlg.Top, frmCmnDlg.pctFileOpen.Width + 2,
frmCmnDlg.pctFileOpen.Height + 2
frmCmnDlg.MOVE (SCREEN.Width - frmCmnDlg.Width) \ 2, ((SCREEN.Height -
frmCmnDlg.Height) \ 2) - 2

' Set dialog colors.
frmCmnDlg.ForeColor = ForeColor
frmCmnDlg.BackColor = BackColor
frmCmnDlg.pctFileOpen.ForeColor = ForeColor
frmCmnDlg.pctFileOpen.BackColor = BackColor
frmCmnDlg.lblOpenFile.ForeColor = ForeColor
frmCmnDlg.lblOpenFile.BackColor = BackColor
frmCmnDlg.txtOpenFile.ForeColor = ForeColor
frmCmnDlg.txtOpenFile.BackColor = BackColor
frmCmnDlg.lblOpenPath.ForeColor = ForeColor
frmCmnDlg.lblOpenPath.BackColor = BackColor
frmCmnDlg.filOpenList.ForeColor = ForeColor
frmCmnDlg.filOpenList.BackColor = BackColor
frmCmnDlg.drvOpenList.ForeColor = ForeColor
frmCmnDlg.drvOpenList.BackColor = BackColor
frmCmnDlg.dirOpenList.ForeColor = ForeColor
frmCmnDlg.dirOpenList.BackColor = BackColor
frmCmnDlg.cmdOpenOK.BackColor = BackColor
frmCmnDlg.cmdOpenCancel.BackColor = BackColor

' Display dialog modally.
frmCmnDlg.SHOW 1

' Determine if user canceled dialog.
IF frmCmnDlg.cmdOpenCancel.Tag <> "FALSE" THEN
    Cancel = TRUE
' If not, return FileName and PathName.
ELSE
    Cancel = FALSE
    FileName = frmCmnDlg.txtOpenFile.Text
    PathName = frmCmnDlg.filOpenList.Path
    frmCmnDlg.cmdOpenCancel.Tag = ""
END IF

' Hide or unload dialog and return control to user's program.
' (Hide if user chose to preload form for performance.)
IF LEFT$(frmCmnDlg.Tag, 1) = "H" THEN
    frmCmnDlg.pctFileOpen.Visible = FALSE
    frmCmnDlg.HIDE
ELSE
    UNLOAD frmCmnDlg
END IF

EXIT SUB

' Option error handling routine.

```

```

' Ignore errors here and let dialog's controls
' handle the errors.
FileOpenError:
    SELECT CASE ERR
        CASE 7:                                     ' Out of memory.
            MSGBOX "Out of memory. Can't load dialog.", 0, "FileOpen"
            Cancel = TRUE
            EXIT SUB
        CASE ELSE
            RESUME NEXT
    END SELECT
END SUB

' FilePrint common dialog support routine
'
' Displays Print dialog which allows users to select
' Print destination (PRINTER.PrintTarget) and the
' number of copies to print. This procedure only provides
' the user interface and returns user input. It does
' not actually carry out the corresponding action.
'
' Parameters:
' Copies - returns the number of copies (1 to 99) the user wants
'           to print. To supply default number of copies
'           in dialog, assign default to Copies then
'           pass it to this procedure (default when Copies
'           is 0 is 1).
' ForeColor - sets the dialog foreground color. Does not affect
'             SCREEN.ControlPanel color settings.
' BackColor - sets the dialog background color. Does not affect
'             SCREEN.ControlPanel color settings.
' Cancel - returns whether or not user pressed the dialog's Cancel
'           button. True (-1) means the user cancelled the dialog.
'
SUB FilePrint (Copies AS INTEGER, ForeColor AS INTEGER, BackColor AS INTEGER, Cancel
AS INTEGER)

    DIM i AS INTEGER

    ON LOCAL ERROR GOTO FilePrintError

    frmCmnDlg.Caption = "Print"                 ' Set form caption.

    ' Determine default number of copies.
    IF Copies = 0 THEN
        frmCmnDlg.txtPrintCopies.Text = "1"
    ELSE
        frmCmnDlg.txtPrintCopies.Text = STR$(Copies)
    END IF

    ' Set default and cancel command buttons.
    frmCmnDlg.cmdPrintOK.Default = TRUE
    frmCmnDlg.cmdPrintCancel.Cancel = TRUE

    ' Size and position Print container.
    frmCmnDlg.pctFilePrint.BorderStyle = 0
    frmCmnDlg.pctFilePrint.Visible = TRUE

    ' Size and center dialog.
    frmCmnDlg.MOVE frmCmnDlg.Left, frmCmnDlg.Top, frmCmnDlg.pctFilePrint.Width + 2,
    frmCmnDlg.pctFilePrint.Height + 2
    frmCmnDlg.MOVE (SCREEN.Width - frmCmnDlg.Width) \ 2, ((SCREEN.Height -
    frmCmnDlg.Height) \ 2) - 2

```

```

' Set dialog colors.
frmCmnDlg.ForeColor = ForeColor
frmCmnDlg.BackColor = BackColor
frmCmnDlg.pctFilePrint.ForeColor = ForeColor
frmCmnDlg.pctFilePrint.BackColor = BackColor
frmCmnDlg.lblPrintCopies.ForeColor = ForeColor
frmCmnDlg.lblPrintCopies.BackColor = BackColor
frmCmnDlg.txtPrintCopies.ForeColor = ForeColor
frmCmnDlg.txtPrintCopies.BackColor = BackColor
frmCmnDlg.txtPrintFile.ForeColor = ForeColor
frmCmnDlg.txtPrintFile.BackColor = BackColor
frmCmnDlg.fraPrintTarget.ForeColor = ForeColor
frmCmnDlg.fraPrintTarget.BackColor = BackColor
FOR i% = 0 TO 3
    frmCmnDlg.optPrintTarget(i%).ForeColor = ForeColor
    frmCmnDlg.optPrintTarget(i%).BackColor = BackColor
NEXT i%
FOR i% = 0 TO 1
    frmCmnDlg.optPrintAppend(i%).ForeColor = ForeColor
    frmCmnDlg.optPrintAppend(i%).BackColor = BackColor
NEXT i%
frmCmnDlg.lblPrintAppend.ForeColor = ForeColor
frmCmnDlg.lblPrintAppend.BackColor = BackColor
frmCmnDlg.cmdPrintOK.BackColor = BackColor
frmCmnDlg.cmdPrintCancel.BackColor = BackColor

' Display dialog modally.
frmCmnDlg.SHOW 1

' Determine if user canceled dialog.
IF frmCmnDlg.cmdPrintCancel.Tag <> "FALSE" THEN
    Cancel = TRUE
' If not, return number of copies to print.
ELSE
    Cancel = FALSE
    IF VAL(frmCmnDlg.txtPrintCopies.Text) > 99 THEN
        Copies = 99
    ELSEIF VAL(frmCmnDlg.txtPrintCopies.Text) < 1 THEN
        Copies = 1
    ELSE
        Copies = VAL(frmCmnDlg.txtPrintCopies.Text)
    END IF
    frmCmnDlg.cmdPrintCancel.Tag = ""
END IF

' Hide or unload dialog and return control to user's program.
' (Hide if user chose to preload form for performance.)
IF LEFT$(frmCmnDlg.Tag, 1) = "H" THEN
    frmCmnDlg.pctFilePrint.Visible = FALSE
    frmCmnDlg.HIDE
ELSE
    UNLOAD frmCmnDlg
END IF

EXIT SUB

' Error handling routine.
FilePrintError:
    SELECT CASE ERR
    CASE 7:                                         ' Out of memory.
        MSGBOX "Out of memory.  Can't load dialog.", 0, "FindPrint"
        Cancel = TRUE

```

```

        EXIT SUB
CASE ELSE
    RESUME NEXT
END SELECT
END SUB

' FileSave common dialog support routine
'
' Displays Save dialog which allows users to specify
' filename for subsequent file save operation.
' This procedure only provides the user interface and
' returns user input. It does not actually carry out
' the corresponding action.
'
' Parameters:
'   FileName - returns the name (without path) of the
'               file for the save operation. To supply
'               default filename in dialog, assign default
'               to FileName then pass it to this procedure.
'   PathName - returns the path (without filename) of
'               the file for the save operation. To supply
'               default path in dialog, assign default to
'               PathName then pass it to this procedure.
'               Note, only pass a valid drive and path. Do
'               not include a filename or file pattern.
'   DefaultExt - sets the default search pattern for the
'                 File Listbox. Default pattern when DefaultExt
'                 is null is "*.*". To specify a different
'                 search pattern (i.e. "*.BAS"), assign new
'                 value to DefaultExt then pass it to this
'                 procedure.
'   DialogTitle - sets the dialog title. Default title
'                 when DialogTitle is null is "Save As". To
'                 specify a different title (i.e. "Save My File"),
'                 assign new value to DialogTitle then pass it to
'                 this procedure.
'   ForeColor - sets the dialog foreground color. Does not affect
'               SCREEN.ControlPanel color settings.
'   BackColor - sets the dialog background color. Does not affect
'               SCREEN.ControlPanel color settings.
'   Flags - unused. Use this to customize dialog action if needed.
'   Cancel - returns whether or not user pressed the dialog's Cancel
'             button. True (-1) means the user cancelled the dialog.
'
SUB FileSave (FileName AS STRING, PathName AS STRING, DefaultExt AS STRING,
DialogTitle AS STRING, ForeColor AS INTEGER, BackColor AS INTEGER, Flags AS INTEGER,
Cancel AS INTEGER)
    ' Set up error handling for option validation.
    ON LOCAL ERROR GOTO FileSaveError

    ' Set form caption.
    IF DialogTitle = "" THEN
        frmCmnDlg.Caption = "Save As"
    ELSE
        frmCmnDlg.Caption = DialogTitle
    END IF
    frmCmnDlg.Tag = frmCmnDlg.Tag + "SAVE"                                ' Set form tag for common
unload procedure.

    ' Determine search pattern for file listbox.
    IF DefaultExt <> "" THEN
        frmCmnDlg.filOpenList.Pattern = DefaultExt
    ELSE

```

```

        frmCmnDlg.filOpenList.Pattern = "*.*"
END IF

' Determine default path.
IF PathName <> "" THEN
    ' If the path ends with a backslash, remove it.
    IF RIGHTS$(PathName, 1) = "\" THEN
        PathName = LEFT$(PathName, LEN(PathName) - 1)
    END IF
    ' Set drive and path for file-system controls.

    ' Set File listbox path.  If PathName is different
    ' than current path, PathChange event will be triggered
    ' which updates Drive listbox drive and Directory listbox path.
    frmCmnDlg.filOpenList.Path = PathName
END IF
' Display current path to the user.
frmCmnDlg.lblOpenPath.Caption = frmCmnDlg.filOpenList.Path

' Determine default filename to display in edit field.
IF FileName <> "" THEN
    frmCmnDlg.txtOpenFile.Text = UCASE$(FileName)
ELSE
    frmCmnDlg.txtOpenFile.Text = frmCmnDlg.filOpenList.Pattern
END IF

' Set default and cancel command buttons.
frmCmnDlg.cmdOpenOK.Default = TRUE
frmCmnDlg.cmdOpenCancel.Cancel = TRUE

' Size and position Open/Save container.
frmCmnDlg.pctFileOpen.BorderStyle = 0
frmCmnDlg.pctFileOpen.Visible = TRUE

' Size and center dialog.
frmCmnDlg.MOVE frmCmnDlg.Left, frmCmnDlg.Top, frmCmnDlg.pctFileOpen.Width + 2,
frmCmnDlg.pctFileOpen.Height + 2
frmCmnDlg.MOVE (SCREEN.Width - frmCmnDlg.Width) \ 2, ((SCREEN.Height -
frmCmnDlg.Height) \ 2) - 2

' Set dialog colors.
frmCmnDlg.ForeColor = ForeColor
frmCmnDlg.BackColor = BackColor
frmCmnDlg.pctFileOpen.ForeColor = ForeColor
frmCmnDlg.pctFileOpen.BackColor = BackColor
frmCmnDlg.lblOpenFile.ForeColor = ForeColor
frmCmnDlg.lblOpenFile.BackColor = BackColor
frmCmnDlg.txtOpenFile.ForeColor = ForeColor
frmCmnDlg.txtOpenFile.BackColor = BackColor
frmCmnDlg.lblOpenPath.ForeColor = ForeColor
frmCmnDlg.lblOpenPath.BackColor = BackColor
frmCmnDlg.filOpenList.ForeColor = ForeColor
frmCmnDlg.filOpenList.BackColor = BackColor
frmCmnDlg.drvOpenList.ForeColor = ForeColor
frmCmnDlg.drvOpenList.BackColor = BackColor
frmCmnDlg.dirOpenList.ForeColor = ForeColor
frmCmnDlg.dirOpenList.BackColor = BackColor
frmCmnDlg.cmdOpenOK.BackColor = BackColor
frmCmnDlg.cmdOpenCancel.BackColor = BackColor

' Display dialog modally.
frmCmnDlg.SHOW 1

```

```

' Determine if user canceled dialog.
IF frmCmnDlg.cmdOpenCancel.Tag <> "FALSE" THEN
    Cancel = TRUE
' If not, return FileName and PathName.
ELSE
    Cancel = FALSE
    FileName = frmCmnDlg.txtOpenFile.Text
    PathName = frmCmnDlg.filOpenList.Path
    frmCmnDlg.cmdOpenCancel.Tag = ""
END IF

' Hide or unload dialog and return control to user's program.
' (Hide if user chose to preload form for performance.)
IF LEFT$(frmCmnDlg.Tag, 1) = "H" THEN
    frmCmnDlg.pctFileOpen.Visible = FALSE
    frmCmnDlg.HIDE
    frmCmnDlg.Tag = "H"           ' Reset tag.
ELSE
    UNLOAD frmCmnDlg
END IF

EXIT SUB

' Option error handling routine.
' Ignore errors here and let dialog's controls
' handle the errors.
FileSaveError:
    SELECT CASE ERR
        CASE 7:                               ' Out of memory.
            MSGBOX "Out of memory.  Can't load dialog.", 0, "FileSave"
            Cancel = TRUE
            EXIT SUB
        CASE ELSE
            RESUME NEXT
    END SELECT
END SUB

```

FILENAME: CMNDLGF.FRM

```
Version 1.00
BEGIN Form frmCmnDlg
    AutoRedraw      = 0
    BackColor       = QBColor(7)
    BorderStyle     = 1
    Caption         = "Common Dialog"
    ControlBox      = -1
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(19)
    Left            = Char(12)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag             = ""
    Top             = Char(3)
    Visible         = -1
    Width           = Char(64)
    WindowState     = 0
BEGIN PictureBox pctFilePrint
    AutoRedraw      = 0
    BackColor       = QBColor(7)
    BorderStyle     = 1
    DragMode        = 0
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(10)
    Left            = Char(0)
    MousePointer    = 0
    TabIndex         = 9
    TabStop          = 0
    Tag             = ""
    Top             = Char(0)
    Visible         = 0
    Width           = Char(60)
BEGIN Frame fraPrintTarget
    BackColor       = QBColor(7)
    Caption         = "Print Target"
    DragMode        = 0
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(8)
    Left            = Char(1)
    MousePointer    = 0
    TabIndex         = 10
    Tag             = ""
    Top             = Char(1)
    Visible         = -1
    Width           = Char(43)
BEGIN OptionButton optPrintTarget
    BackColor       = QBColor(7)
    Caption         = "LPT&1"
    DragMode        = 0
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(1)
    Index           = 0
    Left            = Char(1)
    MousePointer    = 0
    TabIndex         = 11
```

```

        TabStop      = -1
        Tag          = ""
        Top          = Char(0)
        Value        = -1
        Visible      = -1
        Width        = Char(9)
    END
    BEGIN OptionButton optPrintTarget
        BackColor    = QBColor(7)
        Caption      = "LPT&2"
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(1)
        Index        = 1
        Left          = Char(1)
        MousePointer = 0
        TabIndex     = 12
        TabStop      = 0
        Tag          = ""
        Top          = Char(1)
        Value        = 0
        Visible      = -1
        Width        = Char(9)
    END
    BEGIN OptionButton optPrintTarget
        BackColor    = QBColor(7)
        Caption      = "LPT&3"
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(1)
        Index        = 2
        Left          = Char(1)
        MousePointer = 0
        TabIndex     = 13
        TabStop      = 0
        Tag          = ""
        Top          = Char(2)
        Value        = 0
        Visible      = -1
        Width        = Char(9)
    END
    BEGIN OptionButton optPrintTarget
        BackColor    = QBColor(7)
        Caption      = "&File:"
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(1)
        Index        = 3
        Left          = Char(1)
        MousePointer = 0
        TabIndex     = 14
        TabStop      = 0
        Tag          = ""
        Top          = Char(3)
        Value        = 0
        Visible      = -1
        Width        = Char(9)
    END
    BEGIN TextBox txtPrintFile
        BackColor    = QBColor(7)

```

```

        BorderStyle = 1
        DragMode = 0
        Enabled = 0
        ForeColor = QBColor(0)
        Height = Char(3)
        Left = Char(11)
        MousePointer = 0
        MultiLine = 0
        ScrollBars = 0
        TabIndex = 15
        TabStop = -1
        Tag = ""
        Text = ""
        Top = Char(2)
        Visible = -1
        Width = Char(29)
    END
    BEGIN PictureBox pctPrintAppend
        AutoRedraw = 0
        BackColor = QBColor(7)
        BorderStyle = 0
        DragMode = 0
        Enabled = 0
        ForeColor = QBColor(0)
        Height = Char(1)
        Left = Char(0)
        MousePointer = 0
        TabIndex = 16
        TabStop = 0
        Tag = ""
        Top = Char(5)
        Visible = -1
        Width = Char(41)
        BEGIN Label lblPrintAppend
            Alignment = 0
            AutoSize = 0
            BackColor = QBColor(7)
            BorderStyle = 0
            Caption = "If file exists:"
            DragMode = 0
            Enabled = -1
            ForeColor = QBColor(0)
            Height = Char(1)
            Left = Char(1)
            MousePointer = 0
            TabIndex = 17
            Tag = ""
            Top = Char(0)
            Visible = -1
            Width = Char(16)
        END
        BEGIN OptionButton optPrintAppend
            BackColor = QBColor(7)
            Caption = "Append"
            DragMode = 0
            Enabled = -1
            ForeColor = QBColor(0)
            Height = Char(1)
            Index = 0
            Left = Char(17)
            MousePointer = 0
            TabIndex = 18
            TabStop = 0
        END
    END

```

```

        Tag      = ""
        Top     = Char(0)
        Value   = 0
        Visible = -1
        Width   = Char(11)
    END
    BEGIN OptionButton optPrintAppend
        BackColor = QBColor(7)
        Caption   = "&Replace"
        DragMode  = 0
        Enabled   = -1
        ForeColor = QBColor(0)
        Height    = Char(1)
        Index    = 1
        Left     = Char(29)
        MousePointer = 0
        TabIndex  = 19
        TabStop   = -1
        Tag      = ""
        Top     = Char(0)
        Value   = -1
        Visible = -1
        Width   = Char(11)
    END
    END
    BEGIN CommandButton cmdPrintOK
        BackColor = QBColor(7)
        Cancel   = 0
        Caption   = "OK"
        Default  = 0
        DragMode = 0
        Enabled   = -1
        Height   = Char(3)
        Left     = Char(46)
        MousePointer = 0
        TabIndex  = 20
        TabStop   = -1
        Tag      = ""
        Top     = Char(1)
        Visible = -1
        Width   = Char(13)
    END
    BEGIN CommandButton cmdPrintCancel
        BackColor = QBColor(7)
        Cancel   = 0
        Caption   = "Cancel"
        Default  = 0
        DragMode = 0
        Enabled   = -1
        Height   = Char(3)
        Left     = Char(46)
        MousePointer = 0
        TabIndex  = 21
        TabStop   = -1
        Tag      = ""
        Top     = Char(4)
        Visible = -1
        Width   = Char(13)
    END
    BEGIN Label lblPrintCopies
        Alignment = 0
        AutoSize  = 0

```

```

        BackColor      = QBColor(7)
        BorderStyle   = 0
        Caption       = "&Copies:"
        DragMode      = 0
        Enabled       = -1
        ForeColor     = QBColor(0)
        Height        = Char(1)
        Left          = Char(46)
        MousePointer = 0
        TabIndex      = 22
        Tag           = ""
        Top           = Char(8)
        Visible       = -1
        Width         = Char(7)
    END
    BEGIN TextBox txtPrintCopies
        BackColor      = QBColor(7)
        BorderStyle   = 1
        DragMode      = 0
        Enabled       = -1
        ForeColor     = QBColor(0)
        Height        = Char(1)
        Left          = Char(54)
        MousePointer = 0
        MultiLine     = 0
        ScrollBars   = 0
        TabIndex      = 23
        TabStop       = -1
        Tag           = ""
        Text          = ""
        Top           = Char(8)
        Visible       = -1
        Width         = Char(5)
    END
    END
    BEGIN PictureBox pctFileOpen
        AutoRedraw    = 0
        BackColor     = QBColor(7)
        BorderStyle   = 1
        DragMode      = 0
        Enabled       = -1
        ForeColor     = QBColor(0)
        Height        = Char(17)
        Left          = Char(0)
        MousePointer = 0
        TabIndex      = 0
        TabStop       = 0
        Tag           = ""
        Top           = Char(0)
        Visible       = 0
        Width         = Char(50)
    BEGIN Label lblOpenFile
        Alignment     = 0
        AutoSize      = 0
        BackColor     = QBColor(7)
        BorderStyle   = 0
        Caption       = "File &Name:"
        DragMode      = 0
        Enabled       = -1
        ForeColor     = QBColor(0)
        Height        = Char(1)
        Left          = Char(1)
        MousePointer = 0
    END

```

```

TabIndex      = 1
Tag           = ""
Top           = Char(2)
Visible       = -1
Width         = Char(11)
END
BEGIN TextBox txtOpenFile
    BackColor    = QBColor(7)
    BorderStyle  = 1
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(3)
    Left          = Char(12)
    MousePointer = 0
    MultiLine    = 0
    ScrollBars   = 0
    TabIndex     = 2
    TabStop      = -1
    Tag          = ""
    Text          = ""
    Top           = Char(1)
    Visible      = -1
    Width         = Char(23)
END
BEGIN CommandButton cmdOpenOK
    BackColor    = QBColor(7)
    Cancel       = 0
    Caption      = "OK"
    Default      = 0
    DragMode     = 0
    Enabled      = -1
    Height       = Char(3)
    Left          = Char(37)
    MousePointer = 0
    TabIndex     = 7
    TabStop      = -1
    Tag          = ""
    Top           = Char(1)
    Visible      = -1
    Width         = Char(12)
END
BEGIN CommandButton cmdOpenCancel
    BackColor    = QBColor(7)
    Cancel       = 0
    Caption      = "Cancel"
    Default      = 0
    DragMode     = 0
    Enabled      = -1
    Height       = Char(3)
    Left          = Char(37)
    MousePointer = 0
    TabIndex     = 8
    TabStop      = -1
    Tag          = ""
    Top           = Char(4)
    Visible      = -1
    Width         = Char(12)
END
BEGIN DriveListBox drvOpenList
    BackColor    = QBColor(7)
    DragMode     = 0
    Enabled      = -1

```

```

ForeColor      = QBColor(0)
Height        = Char(1)
Left          = Char(19)
MousePointer  = 0
TabIndex      = 5
TabStop       = -1
Tag           = ""
Top           = Char(6)
Visible       = -1
Width         = Char(16)
END
BEGIN FileListBox filOpenList
Archive        = -1
BackColor      = QBColor(7)
DragMode       = 0
Enabled        = -1
ForeColor      = QBColor(0)
Height         = Char(11)
Hidden         = 0
Left           = Char(1)
MousePointer  = 0
Normal         = -1
Pattern        = "*.*"
ReadOnly       = -1
System         = 0
TabIndex      = 4
TabStop       = -1
Tag           = ""
Top           = Char(5)
Visible       = -1
Width         = Char(16)
END
BEGIN Label lblOpenPath
Alignment      = 0
AutoSize       = 0
BackColor      = QBColor(7)
BorderStyle    = 0
Caption        = "C:\"
DragMode       = 0
Enabled        = -1
ForeColor      = QBColor(0)
Height         = Char(1)
Left           = Char(2)
MousePointer  = 0
TabIndex      = 3
Tag           = ""
Top           = Char(4)
Visible       = -1
Width         = Char(33)
END
BEGIN DirListBox dirOpenList
BackColor      = QBColor(7)
DragMode       = 0
Enabled        = -1
ForeColor      = QBColor(0)
Height         = Char(9)
Left           = Char(19)
MousePointer  = 0
TabIndex      = 6
TabStop       = -1
Tag           = ""
Top           = Char(7)
Visible       = -1

```

```

        Width      = Char(16)
      END
    END
END

OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

' -----
' Visual Basic for MS-DOS Common Dialog Toolkit
'
' The Common Dialog Toolkit (CMNDLG.BAS and CMNDLGF.FRM)
' provides support for the following dialogs:
'   FileOpen
'   FileSave
'   FilePrint
'
' Support for each dialog is provided via procedures with
' these same names that create the corresponding dialog
' and return user input to your program. These procedures
' only provide the user interface and return user input.
' They do not actually carry out the corresponding actions
' such as opening the file. Detailed descriptions of
' these procedures are contained in the comment headers
' above each.
'
' All common dialogs are created from the same form (CMNDLGF.FRM) .
' The necessary controls for each dialog are children of
' a container picture box for the dialog. Thus the
' form (CMNDLGF.FRM) contains a picture box with
' appropriate controls for common dialog listed above.
' When a particular common dialog is created and displayed,
' the container picture box for that dialog is made visible
' (thus all controls on that picture box become visible)
' and the form is centered and sized to match the
' container picture box.
'
' To use these common dialogs in your programs, include
' CMNDLG.BAS and CMNDLGF.FRM in your program or use the
' supplied library (CMNDLG.LIB, CMNDLGA.LIB - AltMath version
' for Professional Edition only) and Quick library (CMNDLG.QLB)
' and call the appropriate procedure to invoke the dialog
' you need.
'
' Copyright (C) 1982-1992 Microsoft Corporation
'
' You have a royalty-free right to use, modify, reproduce
' and distribute the sample applications and toolkits provided with
' Visual Basic for MS-DOS (and/or any modified version)
' in any way you find useful, provided that you agree that
' Microsoft has no warranty, obligations or liability for
' any of the sample applications or toolkits.
' -----
'
' Click event procedure for Open/Save dialog Cancel button.
' Makes dialog invisible to return control to FileOpen or FileSave
' procedure (dialog was shown modally).
'
SUB cmdOpenCancel_Click ()
  txtOpenFile.SETFOCUS
  Visible = FALSE
END SUB

```

```

' Click event procedure for Open/Save dialog OK button.
' Determines whether user has selected a file or whether
' path and pattern need to be changed.
'
SUB cmdOpenOK_CClick ()
    DIM OldPattern AS STRING

    ' Set up error handling for directory/drive change errors.
    ON LOCAL ERROR GOTO OKError

    cmdOpenOK.SETFOCUS                                ' Set focus to button, so focus can be
    reset to edit field if needed.

    ' Update Directory listbox path if user single
    ' clicked or used arrow keys in Directory listbox
    ' (only double click automatically changes path).
    dirOpenList.Path = dirOpenList.List(dirOpenList.ListIndex)

    ' If edit field filename does not match File listbox filename
    ' then assign edit field value to File listbox filename
    ' and let it determine if path or pattern need to be
    ' changed.
    IF filOpenList.FileName <> txtOpenFile.Text THEN
        OldPattern$ = filOpenList.Pattern      ' Save old pattern.

        ' Let File listbox control determine if path
        ' or pattern or filename needs to be updated.
        ' PathChange event will be triggered if path needs
        ' updating, PatternChange event will be triggered if
        ' pattern needs updating, and DblClick event will
        ' be triggered if a valid filename has been given.
        filOpenList.FileName = txtOpenFile.Text

        ' If a valid filename was not given (dialog is
        ' still visible to user after DblClick event),
        ' then update the edit field appropriately.
        IF Visible = TRUE THEN
            ' If no pattern change was indicated then either
            ' a new filename was specified for Save dialog
            ' or file was not found for Open dialog.
            IF (INSTR(txtOpenFile.Text, "*") + INSTR(txtOpenFile.Text, "?") < 1)
THEN
                IF INSTR(Tag, "SAVE") THEN
                    CALL filOpenList_DblClick
                ELSE
                    MSGBOX "File not found", 0, Caption
                    filOpenList.Pattern = OldPattern$      ' Restore old File listbox
                search pattern.
                    txtOpenFile.SETFOCUS
                END IF
                ' Pattern change was indicated so just update
                ' textbox with pattern.
                ELSE
                    txtOpenFile.Text = filOpenList.Pattern
                    txtOpenFile.SETFOCUS
                END IF
            END IF
            ' File has been selected by user so close dialog
            ' and return control to FileOpen or FileSave routine.
            ELSE
                CALL filOpenList_DblClick
            END IF
        END IF
    END SUB

```

```

    END IF

OKErrorReturn:
    EXIT SUB

' Drive/Path error handling routine.
OKError:
    MSGBOX ERROR$, 0, Caption           ' Display error message.
    txtOpenFile.SETFOCUS               ' Set focus to edit field so error can be
fixed.
    RESUME OKErrorReturn             ' Exit procedure.
END SUB

' Click event procedure for Print dialog Cancel button.
' Makes dialog invisible to return control to FilePrint
' procedure (dialog was shown modally).
'

SUB cmdPrintCancel_Click ()
    Visible = FALSE
END SUB

' Click event procedure for Print dialog OK button.
' Sets print destination (PRINTER.PrintTarget) and
' makes dialog invisible to return control to FilePrint
' procedure (dialog was shown modally).
'

SUB cmdPrintOK_Click ()
    ' Set up error handling for print to file errors.
    ON LOCAL ERROR GOTO PrintError

    ' Set print target
    IF optPrintTarget(0).Value THEN
        PRINTER.PrintTarget = "LPT1:"           ' Use Basic LPT1 device (colon specifies
this).
    ELSEIF optPrintTarget(1).Value THEN
        PRINTER.PrintTarget = "LPT2:"           ' Use Basic LPT2 device (colon specifies
this).
    ELSEIF optPrintTarget(2).Value THEN
        PRINTER.PrintTarget = "LPT3"            ' No Basic LPT3 device, treat as a
normal file open.
    ELSE
        ' Print target is a file.
        PRINTER.PrintTarget = txtPrintFile.Text
        ' If user specified "Replace" instead of "Append"
        ' option, delete existing file.
        IF optPrintAppend(1).Value THEN KILL txtPrintFile.Text
    END IF
    Visible = FALSE
    cmdPrintCancel.Tag = "FALSE"

    EXIT SUB

' Print to file error handling routine.
' Ignores File Not Found error that occurs when
' deleting a file that does not exist (when user
' chooses "Replace" option).
PrintError:
    RESUME NEXT
END SUB

' Change event procedure for Open/Save dialog Directory listbox.
' Updates the File listbox path to reflect
' the directory change.

```

```

' SUB dirOpenList_Change ()
'   Set up error handling for path errors.
ON LOCAL ERROR GOTO DirChangeError

'   Update file listbox path.
filOpenList.Path = dirOpenList.Path

'   Display new path to the user.
lblOpenPath.Caption = dirOpenList.Path

'   Update text box with search pattern.
txtOpenFile.Text = filOpenList.Pattern

DirChangeErrorReturn:
EXIT SUB

' Path change error handling routine.
DirChangeError:
  MSGBOX ERROR$, 0, Caption           ' Display error message.
  txtOpenFile.SETFOCUS               ' Set focus to edit field so error can be
fixed.
  RESUME DirChangeErrorReturn       ' Exit procedure.
END SUB

' Change event procedure for Open/Save dialog Drive listbox.
' Updates the Directory listbox path to reflect
' the drive change.
'

SUB drvOpenList_Change ()
' Set up error handling for path errors.
ON LOCAL ERROR GOTO DriveChangeError

' Update Dir listbox path.
dirOpenList.Path = drvOpenList.Drive

DriveChangeErrorReturn:
EXIT SUB

' Path change error handling routine.
DriveChangeError:
  MSGBOX ERROR$, 0, Caption           ' Display error message.
  drvOpenList.Drive = dirOpenList.Path ' Reset drive.
  RESUME DriveChangeErrorReturn       ' Exit procedure.
END SUB

' Click event procedure for Open/Save dialog File listbox.
' Selects the file and updates the edit field.
'

SUB filOpenList_Click ()
  txtOpenFile.Text = filOpenList.FileName
END SUB

' Double Click event procedure for Open/Save dialog File listbox.
' File has been selected by the user so make dialog
' invisible to return control to FileOpen or FileSave
' procedure (dialog was shown modally).
'

SUB filOpenList_DblClick ()
  txtOpenFile.SETFOCUS
  Visible = FALSE
  cmdOpenCancel.Tag = "FALSE"
END SUB

```

```

' PathChange event procedure for Open/Save dialog File listbox.
' Updates the Drive listbox drive and Directory
' listbox path to reflect the change.
'
SUB filOpenList_PathChange ()
    ' Set up error handling for path errors.
    ON LOCAL ERROR GOTO FileChangeError

    ' Update drive and path.
    drvOpenList.Drive = filOpenList.Path
    dirOpenList.Path = filOpenList.Path

FileChangeErrorReturn:
    EXIT SUB

' Path change error handling routine.
FileChangeError:
    MSGBOX ERROR$, 0, Caption           ' Display error message.
    drvOpenList.Drive = dirOpenList.Path ' Reset drive.
    filOpenList.Path = dirOpenList.Path  ' Reset path.
    RESUME FileChangeErrorReturn       ' Exit procedure.

END SUB

' PatternChange event procedure for Open/Save dialog File listbox.
' Uppercases search pattern for subsequent display
' in edit field.
'
SUB filOpenList_PatternChange ()
    filOpenList.Pattern = UCASE$(filOpenList.Pattern)
END SUB

' Click event procedure for Print dialog PrintTarget option buttons (control array)
' Handles print target selection.
'
SUB optPrintTarget_Click (Index AS INTEGER)
    ' If file is chosen as print target, enable
    ' filename edit field and append/replace options.
    IF Index = 3 THEN
        txtPrintFile.Enabled = TRUE
        pctPrintAppend.Enabled = TRUE
    ' If LPT1, LPT2, LPT3 is chosen as print target,
    ' disable filename edit field and append/replace options.
    ELSE
        txtPrintFile.Enabled = FALSE
        pctPrintAppend.Enabled = FALSE
    END IF
END SUB

```

**FILENAME: DENSFORM.FRM**

Version 1.00

```
BEGIN Form frmDensity
    AutoRedraw      = 0
    BackColor       = QBColor(3)
    BorderStyle     = 1
    Caption         = "Calculation Status"
    ControlBox      = 0
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(12)
    Left            = Char(27)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag             = ""
    Top             = Char(4)
    Visible         = -1
    Width           = Char(24)
    WindowState     = 0
    BEGIN Frame Frame2
        BackColor      = QBColor(3)
        Caption         = ""
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(3)
        Left            = Char(2)
        MousePointer    = 0
        TabIndex        = 0
        Tag             = ""
        Top             = Char(2)
        Visible         = -1
        Width           = Char(18)
        BEGIN Label lblPercent_Complete
            Alignment      = 0
            AutoSize        = 0
            BackColor       = QBColor(3)
            BorderStyle     = 0
            Caption         = ""
            DragMode        = 0
            Enabled         = -1
            ForeColor       = QBColor(0)
            Height          = Char(1)
            Left            = Char(0)
            MousePointer    = 0
            TabIndex        = 1
            Tag             = ""
            Top             = Char(0)
            Visible         = -1
            Width           = Char(16)
        END
    END
    BEGIN CommandButton cmdClose
        BackColor       = QBColor(3)
        Cancel          = 0
        Caption         = "&Close"
        Default         = 0
        DragMode        = 0
        Enabled         = -1
        Height          = Char(3)
```

```

        Left      = Char(2)
        MousePointer = 0
        TabIndex    = 2
        TabStop     = -1
        Tag         = ""
        Top         = Char(6)
        Visible    = -1
        Width       = Char(18)
    END
    BEGIN Label lblCaption
        Alignment  = 0
        AutoSize   = 0
        BackColor  = QBColor(3)
        BorderStyle = 0
        Caption    = "Percent Complete"
        DragMode   = 0
        Enabled    = -1
        ForeColor  = QBColor(0)
        Height     = Char(1)
        Left       = Char(3)
        MousePointer = 0
        TabIndex   = 3
        Tag        = ""
        Top        = Char(1)
        Visible   = -1
        Width      = Char(16)
    END
END
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

'This form allows the user to monitor the status
'of a calculation, and allows the user to terminate
'the calculation before it has been completed.

SUB cmdClose_Click ()
    'This procedure allows the user to end a regional
    'calculation.

    DIM reply AS INTEGER

    'If a calculation is not in progress then unload the calculate form.

    IF frmDensity.cmdClose.caption = "Continue" THEN
        unload frmDensity
    ELSE
        'If the calculation is in progress ask the user if they really
        'want to leave.

        reply = MSGBOX("Do you really want to stop the present " + _
                      "calculation?", MB_YESNO + MB_DEFBUTTON1, "Close")
        IF (reply = IDYES) THEN
            unload frmDensity
        END IF
    END IF

```

END IF

END SUB

FILENAME: DISCLAIMER.FRM

```
Version 1.00
BEGIN Form frmDisclaimer
    AutoRedraw      = -1
    BackColor       = QBColor(3)
    BorderStyle     = 1
    Caption         = "Welcome to SECPOP90"
    ControlBox      = -1
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(18)
    Left             = Char(10)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag              = ""
    Top              = Char(2)
    Visible         = -1
    Width            = Char(59)
    WindowState     = 0
    BEGIN Label lblNotice
        Alignment      = 0
        AutoSize        = 0
        BackColor       = QBColor(3)
        BorderStyle     = 0
        Caption         = "SECPOP90 calculates estimated population and economic
data about any location in the continental United States. It can also estimate
regional population densities within the continental United States. All estimates
are made using 1990 and 1992 U.S. Bureau of Census data. Population estimates are
made using Block level census data. Economic estimates are made using County level
census data.
Numerous results from this program have been successfully validated against existing
data. However, in no event shall the authors or sponsors of this program be liable
for any damages whatsoever arising from the use of, or inability to use, this
program, including any conclusions drawn from the results. "
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(15)
        Left             = Char(1)
        MousePointer    = 0
        TabIndex        = 0
        Tag              = ""
        Top              = Char(1)
        Visible         = -1
        Width            = Char(55)
    END
END
'This is just your general purpose information and disclaimer message
'which is displayed at the beginning of the program.

SUB Form_KeyPress (KeyAscii AS INTEGER)
    UNLOAD frmDisclaimer
END SUB

SUB Form_LostFocus ()
    UNLOAD frmDisclaimer
```

```
END SUB

SUB lblNotice_Click ()
    UNLOAD frmDisclaimer
END SUB
```

**FILENAME: MAIN.FRM**

```
Version 1.00
BEGIN MDIForm frmMain
    AutoRedraw      = -1
    BackColor       = QBColor(1)
    BorderStyle     = 3
    Caption         = "SECPOP90 - SECtor POPulation and Economic Estimator - Version
2.3"
    ControlBox      = 0
    Enabled         = -1
    ForeColor       = QBColor(11)
    Height          = Char(25)
    Left            = Char(0)
    MousePointer    = 0
    Tag             = ""
    Top             = Char(0)
    Visible         = -1
    Width           = Char(80)
BEGIN Menu mnuSiteName
    Caption         = "&Site"
    Checked        = 0
    Enabled        = -1
    Separator      = 0
    Tag            = ""
    Visible        = -1
BEGIN Menu mnuNew_Sitecom
    Caption         = "&New Site"
    Checked        = 0
    Enabled        = -1
    Separator      = 0
    Tag            = ""
    Visible        = -1
END
BEGIN Menu mnuOpen_Sitecom
    Caption         = "&Open Site"
    Checked        = 0
    Enabled        = -1
    Separator      = 0
    Tag            = ""
    Visible        = -1
END
BEGIN Menu mnuSave_Sitecom
    Caption         = "&Save Site"
    Checked        = 0
    Enabled        = 0
    Separator      = 0
    Tag            = ""
    Visible        = -1
END
BEGIN Menu mnuSave_as_Sitecom
    Caption         = "Save Site &As"
    Checked        = 0
    Enabled        = 0
    Separator      = 0
    Tag            = ""
    Visible        = -1
END
END
BEGIN Menu mnuProblemName
    Caption         = "&Problem"
    Checked        = 0
```

```

Enabled      = -1
Separator    = 0
Tag          = ""
Visible      = -1
BEGIN Menu mnuNew_ProblemCom
    Caption      = "&New Problem"
    Checked     = 0
    Enabled     = -1
    Separator   = 0
    Tag         = ""
    Visible     = -1
END
BEGIN Menu mnuOpen_ProblemCom
    Caption      = "&Open Problem"
    Checked     = 0
    Enabled     = -1
    Separator   = 0
    Tag         = ""
    Visible     = -1
END
BEGIN Menu mnuSave_ProblemCom
    Caption      = "&Save Problem"
    Checked     = 0
    Enabled     = 0
    Separator   = 0
    Tag         = ""
    Visible     = -1
END
BEGIN Menu mnuSave_As_ProblemCom
    Caption      = "Save Problem &As"
    Checked     = 0
    Enabled     = 0
    Separator   = 0
    Tag         = ""
    Visible     = -1
END
END
BEGIN Menu mnuCalculateName
    Caption      = "&Calculate"
    Checked     = 0
    Enabled     = -1
    Separator   = 0
    Tag         = ""
    Visible     = -1
BEGIN Menu mnuSite_SpecificCom
    Caption      = "&Site Specific"
    Checked     = 0
    Enabled     = -1
    Separator   = 0
    Tag         = ""
    Visible     = -1
END
BEGIN Menu mnuRegionalCom
    Caption      = "&Regional"
    Checked     = 0
    Enabled     = -1
    Separator   = 0
    Tag         = ""
    Visible     = -1
END
END
BEGIN Menu mnuOutputName
    Caption      = "&Results"

```

```

Checked      = 0
Enabled      = -1
Separator    = 0
Tag          = ""
Visible      = -1
BEGIN Menu mnuRosetteCom
    Caption      = "&Population Rosette"
    Checked      = 0
    Enabled      = -1
    Separator    = 0
    Tag          = ""
    Visible      = -1
END
BEGIN Menu mnuTableCom
    Caption      = "Population &Tables"
    Checked      = 0
    Enabled      = -1
    Separator    = 0
    Tag          = ""
    Visible      = -1
    BEGIN Menu mnuTable_1com
        Caption      = "&Population"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
    BEGIN Menu mnuTable_2com
        Caption      = "&Cumulative"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
END
BEGIN Menu mnuMACCScom
    Caption      = "&MACCS Input File"
    Checked      = 0
    Enabled      = -1
    Separator    = 0
    Tag          = ""
    Visible      = -1
    BEGIN Menu mnuMACCS_Display
        Caption      = "&Display"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
    BEGIN Menu mnuMACCS_Input_FileCom
        Caption      = "&Print"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
END
BEGIN Menu mnuRegional_Displaycom
    Caption      = "&Regional"

```

```

        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
    BEGIN Menu mnuSave_Resultscom
        Caption      = "&Save Results"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
    BEGIN Menu mnuPreviouscom
        Caption      = "&Load Previous Results"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
    BEGIN Menu mnuSetupcom
        Caption      = "Set&up"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
    BEGIN Menu mnuExitname
        Caption      = "E&xit"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    BEGIN Menu mnuDos_Shellcom
        Caption      = "&DOS Shell"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
    BEGIN Menu mnuExitcom
        Caption      = "E&xit"
        Checked      = 0
        Enabled      = -1
        Separator    = 0
        Tag          = ""
        Visible      = -1
    END
    END
END
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

'Data for radii to be used for the regional calculations.

DATA 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

```

```

DATA 11, 12, 13, 14, 15, 16, 17, 18, 19, 20
DATA 30, 40, 50, 60, 70, 80, 90
DATA 100, 110, 120, 130, 140, 150, 160, 170

'Data for forms which display the compass points.

DATA "N", "NNE", "NE", "ENE", "E", "ESE", "SE", "SSE", "S", "SSW", "SW"
DATA "WSW", "W", "WNW", "NW", "NNW"

SUB Form_Load ()
    'Define variables local to this event procedure.

    DIM i AS INTEGER, filenum AS INTEGER, tempstring AS STRING

    'Set Screen attributes for main program screen.

    SCREEN.ControlPanel(DESKTOP_PATTERN) = KEY_SPACE
    SCREEN.ControlPanel(ACTIVE_BORDER_BACKCOLOR) = BLUE
    SCREEN.ControlPanel(ACTIVE_BORDER_FORECOLOR) = BRIGHT_WHITE
    SCREEN.ControlPanel(TITLEBAR_BACKCOLOR) = MAGENTA
    SCREEN.ControlPanel(TITLEBAR_FORECOLOR) = BRIGHT_WHITE
    SCREEN.ControlPanel(MENU_BACKCOLOR) = WHITE
    SCREEN.ControlPanel(MENU_FORECOLOR) = BLACK
    SCREEN.ControlPanel(SCROLLBAR_BACKCOLOR) = CYAN
    SCREEN.ControlPanel(SCROLLBAR_FORECOLOR) = BRIGHT_CYAN
    SCREEN.ControlPanel(ACTIVE_WINDOW_SHADOW) = FALSE

    'Initialize to zero the radial distance array.

    FOR i = 1 TO max_number_of_radii
        radial_distance(i) = 0
    NEXT i

    'Reset the number of radii to 0

    number_of_radii = 0

    'Initialize the radii used for regional calculations.

    restore
    for i = 1 TO max_number_of_radii
        read regional_radii(i)
    next i

    'Read in the direction names (compass points).

    FOR i = 1 TO number_of_segments
        READ directions(i)
    NEXT i

    'Set up error handler for errors in the configuration
    'file, and load in the various parameters from the
    'setup file into the setup form. These parameters
    'are set by the user in the exit menu under setup.
    'Because input will not interact directly with forms,
    'tempstring is used as a buffer for file data.

    'Load in the forms that we need to keep permanently in memory.

    LOAD frmProblem_Data
    LOAD frmSite_Data
    LOAD frmRegion

```

```

LOAD frmSetup

frmDisclaimer.SHOW MODELESS
ON LOCAL ERROR GOTO configerr

'Check and see if config file exists.

IF LEN(DIR$("SECPOP90.CFG")) <> 0 THEN

    'Open file and read in the data from it.

    filenum = FREEFILE
    OPEN "SECPOP90.CFG" FOR INPUT AS filenum
    INPUT #filenum, tempstring
    frmSetup.txtSite_Path.text = tempstring
    INPUT #filenum, tempstring
    frmSetup.txtProblem_Path.text = tempstring
    INPUT #filenum, tempstring
    frmSetup.txtOutput_path.text = tempstring
    INPUT #filenum, tempstring
    frmSetup.txtData_path.text = tempstring
    INPUT #filenum, tempstring
    IF (tempstring = "MACCS") THEN
        frmSetup.optMACCS.value = TRUE
    ELSE
        frmSetup.optCSV.value = TRUE
    END IF
    CLOSE #filenum

    'else if config file doesn't exist use defaults.

ELSE

    'Defaults set in form.

END IF

'Turn off error handler.

ON LOCAL ERROR GOTO 0

EXIT SUB

configerr:

    'If config file format is screwed up use defaults.

    CLOSE filenum
    EXIT SUB

END SUB

SUB mnuDOS_Shellcom_Click ()

    'Remove all forms from the screen.

    UNLOAD frmDisclaimer

    SCREEN.HIDE
    CLS
    LOCATE 1, 1

    'Place shell message on screen.

```

```

SHELL "ECHO Type EXIT to return to SECPOP90"
LOCATE 1, 1

'Run the DOS shell.

SHELL

'Redisplay screen.

SCREEN.SHOW

END SUB

SUB mnuExitCom_Click ()

    DIM reply AS INTEGER

    'Ask the user if they really want to exit.

    reply = MSGBOX("Do you really want to exit?", MB_YESNO + MB_DEFBUTTON1, "Exit")

    IF (reply = IDYES) THEN
        END
    END IF

END SUB

SUB mnuMACCS_Display_Click ()

    'This function creates a temporary file, and then
    'writes into that file the information for the user
    'to view on the screen. This file is then sent to a
    'text form, where the user can view it and scroll
    'the various parts of the data.

    DIM filenum AS INTEGER

    UNLOAD frmDisclaimer

    'Create temporary file, and tell the print_MACCS
    'input_file function to send the info to this file.

    PRINTER.PrintTarget = "$$TEMP$$.$$$"

    'Write the actual data to file.

    CALL print_MACCS_input_file

    'Reset to printer the destination of info generated
    'by the print_MACCS_input function.

    PRINTER.PrintTarget = ""

    filenum = FREEFILE

    'Display information in text form.

    OPEN "$$TEMP$$.$$$" FOR INPUT AS filenum
    frmOutput.txtOutput.text = ""
    frmOutput.txtOutput.text = INPUT$(LOF(filenum), filenum)
    frmOutput.SHOW MODAL

```

```

'Close and delete temporary file.

CLOSE filenum

KILL "$$TEMP$$.$$$"

END SUB

SUB mnuMACCS_Input_FileCom_Click ()

'This function prints the data in MACCS Site File format.

'These variables are used to set up the print dialog.

DIM ForeColor AS INTEGER, BackColor AS INTEGER
DIM Copies AS INTEGER, Cancel AS INTEGER, i AS INTEGER

DIM site_name AS STRING, site_remarks AS STRING

UNLOAD frmDisclaimer

'Set colors for Dialog box

BackColor = WHITE
ForeColor = BLACK

'Call the standard print dialog.

CALL FilePrint(Copies, ForeColor, BackColor, Cancel)

'If user did not select cancel in the print dialog then print.

IF NOT Cancel THEN

  'Set up error handling for print errors.

  on local error goto print_error

  'If the user is printing to a printer adjust the margins so that
  'there are no extra line feeds after the lines that have 80
  'columns otherwise prepend the output path to the printer
  'target.

  if((printer.PrintTarget = "LPT1:") or
     (printer.PrintTarget = "LPT2:") or
     (printer.PrintTarget = "LPT3:")) then

    width printer.PrintTarget, 81

  else

    printer.PrintTarget = frmSetup.txtOutput_path.text + _
      "\\" + printer.PrintTarget

  end if

  'Print however many copies specified, each followed by a form feed.

  FOR i = 1 TO Copies
    CALL print_MACCS_input_file
    PRINTER.NEWPAGE
  NEXT i
  PRINTER.ENDDOC

```

```

END IF

EXIT SUB

print_error:
    'Deal with errors such as printer is off line or file not found.

    MSGBOX "Device error - check printer or other print device."
    EXIT SUB

END SUB

SUB mnuNew_ProblemCom_Click ()
    'This function asks the user if they want to review
    'previously entered data or create a new problem.

    DIM i AS INTEGER, reply AS INTEGER

    UNLOAD frmDisclaimer

    'If no problem has already been loaded, drop user
    'directly into the new problem form.

    IF frmProblem_Data.caption = "" THEN
        CALL new_problem

        'Otherwise ask user if they want to review the old
        'data.  If they do, then show it, otherwise call
        'the new problem form.

    ELSE
        reply = MSGBOX("Do you wish to review the previous problem data?",

MB_YESNOCANCEL, "New Problem")
        IF reply = IDYES THEN
            frmProblem_Data.SHOW MODAL
        ELSEIF reply = IDNO THEN
            CALL new_problem
        END IF

    END IF

END SUB

SUB mnuNew_SiteCom_Click ()
    'This function asks the user if they want to review
    'previous data or create a new site.

    DIM reply AS INTEGER

    UNLOAD frmDisclaimer

    'If no data has been entered, go directly to new site form.

    IF frmSite_Data.caption = "" THEN
        CALL new_site

```

```

ELSE

    'If data has been entered, ask the user if they want to see it.
    'If yes, then show old data, otherwise go to new site form.

        reply = MSGBOX("Do you wish to review the previous site data?", 
MB_YESNOCANCEL, "New Site")
        IF reply = IDYES THEN
            frmSite_Data.SHOW MODAL
        ELSEIF reply = IDNO THEN
            CALL new_site
        END IF

    END IF

END SUB

SUB mnuOpen_ProblemCom_Click ()

    'This subroutine allows the user to look at a previously
    'created problem data or review problem data that has already
    'been loaded.

    DIM reply AS INTEGER

    UNLOAD frmDisclaimer

    'If data has not yet been entered, go directly to the
    'open problem function.

    IF frmProblem_Data.caption = "" THEN
        CALL open_problem

    'Otherwise, ask the user if they want to review the
    'previous problem data.  If so, show it to them, otherwise
    'call open problem.

    ELSE

        reply = MSGBOX("Do you wish to review the previous problem data?", 
MB_YESNOCANCEL, "Open Problem File")
        IF reply = IDYES THEN
            frmProblem_Data.SHOW MODAL
        ELSEIF reply = IDNO THEN
            CALL open_problem
        END IF

    END IF

END SUB

SUB mnuOpen_SiteCom_Click ()

    'This subroutine allows the user to look at a previously
    'created site data or review site data that has already
    'been loaded.

    DIM reply AS INTEGER

    UNLOAD frmDisclaimer

    'If data has not yet been entered, go directly to the

```

```

'open site function.

IF frmSite_Data.caption = "" THEN
    CALL open_site

'Otherwise, ask the user if they want to review the
'previous site data.  If so, show it to them, otherwise
'call open site.

ELSE

    reply = MSGBOX("Do you wish to review the previous site data?", 
MB_YESNOCANCEL, "Open Site File")
    IF reply = IDYES THEN
        frmSite_Data.SHOW MODAL
    ELSEIF reply = IDNO THEN
        CALL open_site
    END IF

END IF

END SUB

SUB mnuPreviouscom_Click ()

    'Load in previously saved output data so that the user
    'can view it again, in any of the standard forms.

    DIM tempinteger AS INTEGER, tempsingle AS SINGLE
    DIM filenum AS INTEGER, i AS INTEGER, j AS INTEGER, k AS INTEGER
    DIM tempstring AS STRING
    DIM ForeColor AS INTEGER, BackColor AS INTEGER
    DIM Flags AS INTEGER, Cancel AS INTEGER
    DIM FileName AS STRING
    STATIC PathName AS STRING
    DIM DefaultExt AS STRING, DialogTitle AS STRING

    UNLOAD frmDisclaimer

    'Set defaults for the open file dialog box.

    DefaultExt = "*.TXT"
    DialogTitle = "Load Previous Data"
    BackColor = WHITE
    ForeColor = BLACK
    PathName = frmSetup.txtOutput_path.text

    'Set up error checking.

    ON LOCAL ERROR GOTO previous_error

    'Prompt the user for a filename, and if they do not
    'cancel the operation, open the file using the
    'first available file handle.

    CALL FileOpen(FileName, PathName, DefaultExt, DialogTitle, ForeColor, BackColor,
Flags, Cancel)

    'If the user did not select cancel from the previous dialogue box,
    'then load the file.

    IF NOT Cancel THEN

```

```

'Determine first available file handle.

filenum = FREEFILE
IF PathName <> "" THEN FileName = PathName + "\" + FileName
OPEN FileName FOR INPUT AS filenum

'Read in the header information.

tempstring = input$(40, #filenum)
LINE INPUT #filenum, tempstring
frmSite_data.txtSite_name.Text = tempstring

tempstring = input$(6, #filenum)
tempstring = input$(2, #filenum)
frmSite_data.txtLatitude_degrees.Text = ltrim$(tempstring)
tempstring = input$(1, #filenum)
tempstring = input$(2, #filenum)
frmSite_data.txtLatitude_minutes.Text = ltrim$(tempstring)
tempstring = input$(1, #filenum)
tempstring = input$(2, #filenum)
frmSite_data.txtLatitude_seconds.Text = ltrim$(tempstring)

tempstring = input$(9, #filenum)
tempstring = input$(3, #filenum)
frmSite_data.txtLongitude_degrees.Text = ltrim$(tempstring)
tempstring = input$(1, #filenum)
tempstring = input$(2, #filenum)
frmSite_data.txtLongitude_minutes.Text = ltrim$(tempstring)
tempstring = input$(1, #filenum)
tempstring = input$(2, #filenum)
frmSite_data.txtLongitude_seconds.Text = ltrim$(tempstring)

tempstring = input$(26, #filenum)
input #filenum, tempsingle
LINE INPUT #filenum, tempstring
frmProblem_data.txtPopulation_multiplier.Text = str$(tempsingle)

'Pull in various numbers and strings from the file,
'discarding extraneous labels and information.

INPUT #filenum, tempsingle
number_of_radii = tempsingle

'Discard next 5 lines from file

FOR i = 1 TO 5 STEP 1
    LINE INPUT #filenum, tempstring
NEXT i

'Determine number of economic regions.

INPUT #filenum, tempsingle
number_econ_regions = tempsingle
LINE INPUT #filenum, tempstring

'Fill number of radii array from file data.

LINE INPUT #filenum, tempstring

FOR i = 1 TO number_of_radii
    INPUT #filenum, tempsingle
    radial_distance(i) = tempsingle

```

```

NEXT i

'Fill total population array for each segment and radii.

LINE INPUT #filenum, tempstring

FOR i = 1 TO number_of_segments
    FOR j = 1 TO number_of_radii
        INPUT #filenum, tempsingle
        sector_population(i, j) = tempsingle
    NEXT j
NEXT i

'Fill fraction land array for each segment and radii.

LINE INPUT #filenum, tempstring

FOR i = 1 TO number_of_segments
    FOR j = 1 TO number_of_radii
        INPUT #filenum, tempsingle
        sector_frlnd(i, j) = tempsingle
    NEXT j
NEXT i

'Fill region index array.

LINE INPUT #filenum, tempstring

FOR i = 0 TO (number_of_segments - 1)
    FOR j = 1 TO number_of_radii
        tempstring = INPUT$(2, #filenum)
        IF VAL(tempstring) = 0 THEN
            tempstring = INPUT$(2, #filenum)
        END IF
        region_index(j, i) = VAL(tempstring)
    NEXT j
NEXT i
LINE INPUT #filenum, tempstring

'Suck in and throw away the dummy watershed indices.

LINE INPUT #filenum, tempstring
FOR i = 1 TO number_of_segments
    LINE INPUT #filenum, tempstring
NEXT i

'Suck in and throw away the dummy crop season and share info.

LINE INPUT #filenum, tempstring
LINE INPUT #filenum, tempstring

'Suck in and throw away the dummy watershed definition.

LINE INPUT #filenum, tempstring
LINE INPUT #filenum, tempstring

'Fill regional data array structs.

LINE INPUT #filenum, tempstring
FOR i = 1 TO number_econ_regions
    INPUT #filenum, tempsingle
    tempstring = INPUT$(10, #filenum)
    INPUT #filenum, tempsingle

```

```

        econ_data(i).region_frmfrc = tempsingle
        INPUT #filenum, tempsingle
        econ_data(i).region_dpf = tempsingle
        INPUT #filenum, tempsingle
        econ_data(i).region_asfp = tempsingle
        INPUT #filenum, tempsingle
        econ_data(i).region_vfrm = tempsingle
        INPUT #filenum, tempsingle
        econ_data(i).region_vnfrm = tempsingle
    NEXT i

END IF

'Update site and problem form captions and remarks.

frmSite_data.Caption = filename
frmSite_data.txtSite_remarks.Text = ""
frmProblem_data.Caption = filename
frmProblem_data.txtProblem_remarks.Text = ""

'Update problem form radial distances, units, and number of regions.

frmProblem_data.txtRadial_distance1.Text = str$(radial_distance(1))
frmProblem_data.txtRadial_distance2.Text = str$(radial_distance(2))
frmProblem_data.txtRadial_distance3.Text = str$(radial_distance(3))

frmProblem_Data.optKilometers.value = TRUE

frmProblem_Data.lblNumber_of_regions.Caption = _
    str$(number_econ_regions)

EXIT SUB

previous_error:

'If there is an error, identify type if possible,
'inform the user of the problem, and leave subroutine.

IF (ERR = 62) THEN
    MSGBOX "Error: Attempt has been made to read past the end of file.  Data
file has been ccorrupted."
ELSEIF (ERR = 53) OR (ERR = 75) OR (ERR = 76) OR (ERR = 55) THEN
    MSGBOX "Invalid filename or path, or file access denied"
ELSEIF (ERR = 61) OR (ERR = 68) OR (ERR = 71) OR (ERR = 72) THEN
    MSGBOX "The disk drive is not ready, or a file error has occured"
ELSE MSGBOX Error$(err)
END IF
EXIT SUB

END SUB

SUB mnuRegionalCom_Click ()
    'Call the forms and functions which handle the
    'regional calculations

    UNLOAD frmDisclaimer

    'load form which prompts user for certain input
    'necessary for regional calculations

    LOAD frmMake_a_Circle
    frmMake_a_Circle.SHOW MODAL

```

```

END SUB

SUB mnuRegional_Displaycom_Click ()
    UNLOAD frmDisclaimer
    'Display regional calculation results.
    RUN "run_mppr.exe"
END SUB

SUB mnuRosetteCom_Click ()
    UNLOAD frmDisclaimer
    'Draw the Rosette graph.
    CALL Rosette
END SUB

SUB mnuSave_As_ProblemCom_Click ()
    'This event procedure calls the save as problem subroutine
    'which saves the problem data with a new file name.

    UNLOAD frmDisclaimer
    CALL save_as_problem
END SUB

SUB mnuSave_As_SiteCom_Click ()
    'This subroutine calls the save as site subroutine which saves
    'the site data with a new file name.

    UNLOAD frmDisclaimer
    CALL save_as_site
END SUB

SUB mnuSave_ProblemCom_Click ()
    'This subroutine calls the save problem subroutine
    'which saves the problem data.

    UNLOAD frmDisclaimer
    CALL save_problem
END SUB

SUB mnuSave_Resultscom_Click ()
    'This event procedure saves the results so that they are not lost when
    'the program is exited. For this function to work, a problem must be
    'loaded, along with the data that corresponds to that problem

    DIM filenum AS INTEGER

```

```

DIM ForeColor AS INTEGER, BackColor AS INTEGER
DIM Flags AS INTEGER, Cancel AS INTEGER
DIM FileName AS STRING, PathName AS STRING
DIM DefaultExt AS STRING, DialogTitle AS STRING

UNLOAD frmDisclaimer

'Initialize dialoge box which asks for file name

PathName = frmSetup.txtOutput_path.text
DefaultExt = "*.TXT"
DialogTitle = "Save Data"

'If problem is not loaded, tell user and exit subroutine

IF frmProblem_Data.txtradiial_distance1.text = "" THEN
    MSGBOX "Data is not yet available - Site file has not been loaded"
    EXIT SUB
END IF

'Prompt user for filename

CALL FileSave(FileName, PathName, DefaultExt, DialogTitle, 7, 1, Flags, Cancel)

'If user does not select cancel from dialoge box, proceed

IF NOT Cancel THEN

    'If file is not in current directory, append path to file

    IF PathName <> "" THEN FileName = PathName + "\\" + FileName

    'Enable local error checking

    ON LOCAL ERROR GOTO save_results_error

    'If file already exists, delete it (to prevent appending to it)

    IF (LEN(DIR$(FileName)) <> 0) THEN KILL FileName

    'Reset printer object so that it points to the specified file

    PRINTER.PrintTarget = FileName

    'Call print function to print maccs data to the file

    CALL print_MACCS_input_file

    'Reset printer object so that it points to the printer

    PRINTER.PrintTarget = ""

END IF

'Turn off local error checking

ON LOCAL ERROR GOTO 0

EXIT SUB

save_results_error:
    'If any kind of error occurs, inform the user and exit subroutine

```

```

MSGBOX "A file error has occurred. Please check all filenames and devices"
EXIT SUB

END SUB

SUB mnuSave_SiteCom_Click ()
    'This subroutine calls the save site subroutine
    'which saves the site data.

    UNLOAD frmDisclaimer

    CALL save_site

END SUB

SUB mmusetupcom_Click ()
    'This subroutine shows the setup form. This
    'form allows the user to set up paths and file
    'saving formats.

    UNLOAD frmDisclaimer

    frmSetup.SHOW MODAL

END SUB

SUB mnuSite_specificCom_Click ()
    UNLOAD frmDisclaimer

    'This subroutine checks to see if there is enough data to perform
    'site calculations, and if so, calls the form to do the actual math.
    'For further documentation, see frmCalculate.

    'Verify that there are radial distances and coordinates.

    IF ((number_of_radii = 0) OR (VAL(frmProblem_Data.txtPopulation_multiplier.text)
    = 0)) THEN
        MSGBOX "Unable to calculate populations." + CHR$(KEY_RETURN) + "Problem Data
        form is incomplete.", MB_OK, "Calculation Error"
    ELSEIF ((VAL(frmSite_Data.txtLongitude_Degrees.text) = 0) OR
    (VAL(frmSite_Data.txtLatitude_Degrees.text) = 0)) THEN
        MSGBOX "Unable to calculate populations." + CHR$(KEY_RETURN) + " Site Data
        form is incomplete.", MB_OK, "Calculation Error"
    ELSE
        LOAD frmCalculate
        frmCalculate.SHOW MODAL
    END IF

END SUB

SUB mnuTable_1com_Click ()
    'This event procedure allows the user to view data
    'in tabular format by showing the first table
    'form. This form must be loaded and unloaded each
    'time that it is displayed so that the values
    'displayed will be updated. For further documentation
    'see frmTable_1.

```

```
UNLOAD frmDisclaimer

table_type = 1
LOAD frmTable_1
frmTable_1.SHOW MODAL

END SUB

SUB mnuTable_2com_Click ()

'This event procedure allows the user to view data
'in tabular format by showing the second table
'form.  This form must be loaded and unloaded each
'time that it is displayed so that the values
'displayed will be updated.  For further documentation
'see frmTable_1.

UNLOAD frmDisclaimer

table_type = 2
LOAD frmTable_1
frmTable_1.SHOW MODAL

END SUB
```

**FILENAME: OUTPCODE.BAS**

```
option explicit

'$include: 'secpop90.bi'

'This module contains the print_MACCS_input_file subroutine which is
'used to either print or save calculation results in a format
'compatible with the MACCS requirements for a site input file or as a
'quotation mark and comma delimited file suitable for import into most
'spreadsheets.

sub print_MACCS_input_file ()

    'This routine prints SECPOP90 output in MACCS compatible form or in
    'a quotation mark comma delimited form. It can be used to print to
    'a printer or a file.

    dim quote as string, comma as string
    dim blank as string, degree as string
    dim i as integer, j as integer

    'Set quote and comma strings to null if MACCS input file format is
    'selected or to "" and ',' if comma separated variable (CSV) format
    'is selected.

    if (frmSetup.optMACCS.Value) then
        quote = ""
        comma = ""
        blank = " "
    elseif (frmSetup.optCSV.Value) then
        quote = """
        comma = ","
        blank = ""
    else
        quote = ""
        comma = ""
        blank = " "
    end if

    'Set degree character.

    degree = chr$(248)

    'Enable local error checking for things like printer off line, etc.

    on local error goto print_maccs_error

    'Initialize RADDIS array to Kilometers.

    for i = 1 to number_of_radii step 1

        if (frmProblem_Data.optKilometers.value = TRUE) then
            raddis(i) = radial_distance(i)
        else
            raddis(i) = radial_distance(i) * miles_to_kilometers
```

```

    end if

next 1

'Print (or save) results.

'Print the first line.

printer.print quote; blank;
printer.print "SECPOP90 V2.3 ";
if (frmSetup.optMACCS.Value) then
    printer.print "MACCS ";
elseif (frmSetup.optCSV.Value) then
    printer.print "CSV ";
else
    printer.print "MACCS ";
end if
printer.print "Site Data File for";
printer.print quote; comma; blank;

printer.print quote;
printer.print ltrim$(rtrim$(frmSite_data.txtSite_name.Text));
printer.print quote

'Print the second line.

printer.print quote; blank;
printer.print "Lat:";
printer.print quote; comma; blank;

printer.print using "##";
    val(frmSite_data.txtLatitude_degrees.Text);
printer.print comma;

printer.print quote;
printer.print degree;
printer.print quote; comma;

printer.print using "##";
    val(frmSite_data.txtLatitude_minutes.Text);
printer.print comma;

printer.print quote;
printer.print "'";
printer.print quote; comma;

printer.print using "##";
    val(frmSite_data.txtLatitude_seconds.Text);
printer.print comma;

printer.print quote;
printer.print "'";
printer.print quote; comma; blank;

printer.print quote;
printer.print "Long:";
printer.print quote; comma; blank;

printer.print using "###";
    val(frmSite_data.txtLongitude_degrees.Text);

```

```

printer.print comma;

printer.print quote;
printer.print degree;
printer.print quote; comma;

printer.print using "##"; _
    val(frmSite_data.txtLongitude_minutes.Text);
printer.print comma;

printer.print quote;
printer.print "";
printer.print quote; comma;

printer.print using "##"; _
    val(frmSite_data.txtLongitude_seconds.Text);
printer.print comma;

printer.print quote;
printer.print "'";
printer.print quote; comma; blank;

printer.print quote;
printer.print "Population multiplier:";
printer.print quote; comma; blank;

printer.print using "####.####"; _
    val(frmProblem_data.txtPopulation_multiplier.Text);
printer.print comma; blank; blank; blank;

printer.print quote;
printer.print format$(now, "mm/dd/yyyy");
printer.print quote

'Print the data block sizes.

printer.print using "####"; number_of_radii;
printer.print comma; blank;

printer.print quote;
printer.print "SPATIAL INTERVALS";
printer.print quote

printer.print using "####"; number_of_segments;
printer.print comma; blank;

printer.print quote;
printer.print "WIND DIRECTIONS";
printer.print quote

printer.print using "####"; 1;
printer.print comma; blank;

printer.print quote;
printer.print "CROP CATEGORIES";
printer.print quote

printer.print using "####"; 1;
printer.print comma; blank;

printer.print quote;
printer.print "WATER PATHWAY ISOTOPES";

```

```

printer.print quote

printer.print using "####"; 1;
printer.print comma; blank;

printer.print quote;
printer.print "WATERSHEDS";
printer.print quote

printer.print using "####"; number_econ_regions;
printer.print comma; blank;

printer.print quote;
printer.print "ECONOMIC REGIONS";
printer.print quote

'Print the geometry data block.

printer.print quote; blank;
printer.print "SPATIAL DISTANCES      KILOMETERS";
printer.print quote

for i = 1 to number_of_radii step 1

    printer.print using "#####.##"; raddis(i);
    printer.print comma;

    if ((i mod 8) = 0) then
        printer.print
    end if

next i

'Go to the next line if necessary.

if ((number_of_radii mod 8) <> 0) then
    printer.print
end if

'Print the population data block.

printer.print quote; blank;
printer.print "POPULATION";
printer.print quote

for i = 1 to number_of_segments step 1

    for j = 1 to number_of_radii step 1

        printer.print using "#####.##"; sector_population(i, j);
        printer.print comma;

        if ((j mod 8) = 0) then
            printer.print
        end if

    next j

    'Go to the next line if necessary.

    if ((number_of_radii mod 8) <> 0) then

```

```

        printer.print
    end if

next i

'Print the land fraction block.

printer.print quote; blank;
printer.print "LAND FRACTION";
printer.print quote

for i = 1 to number_of_segments step 1

    for j = 1 to number_of_radii step 1

        printer.print using "##.##"; sector_frcInd(i, j);
        printer.print comma;

        if ((j mod 16) = 0) then
            printer.print
        end if

    next j

    'Go to the next line if necessary.

    if ((number_of_radii mod 16) <> 0) then
        printer.print
    end if

next i

'Print the region identification block.

printer.print quote; blank;
printer.print "REGION INDEX";
printer.print quote

for i = 0 to (number_of_segments - 1) step 1

    for j = 1 to number_of_radii step 1

        printer.print using "##"; region_index(j, i);
        printer.print comma;

    next j

    printer.print

next i

'Print the (dummy) watershed identification block.

printer.print quote; blank;
printer.print "WATERSHED INDEX";
printer.print quote

for i = 1 to number_of_segments step 1

    for j = 1 to number_of_radii step 1

```

```

        printer.print using "##"; 1;
        printer.print comma;

    next ]

    printer.print

next i

'Print the (dummy) agricultural data block.

printer.print quote; blank;
printer.print "CROP SEASON AND SHARE";
printer.print quote

printer.print using "####"; 1;
printer.print comma;

printer.print quote; blank;
printer.print "NONE" ;
printer.print quote; comma;

printer.print using "####."; 1.0;
printer.print comma;

printer.print using "####."; 1.0;
printer.print comma;

printer.print using "#####.#####"; 0.0

'Print the (dummy) watershed data block.

printer.print quote; blank;
printer.print "WATERSHED DEFINITION";
printer.print quote

printer.print using "####"; 1;
printer.print comma;

printer.print quote; blank;
printer.print "NONE" ;
printer.print quote; comma;

printer.print using "#####.##"; 0.0;
printer.print comma;

printer.print using "#####.##"; 0.0;
printer.print comma;

printer.print using "#####.##"; 0.0;
printer.print comma;

printer.print using "#####.##"; 0.0

```

'Print the regional economic data block.

```

printer.print quote; blank;
printer.print "REGIONAL ECONOMIC DATA";
printer.print quote

```

```

for i = 1 to number_econ_regions step 1

    printer.print using "####"; i;
    printer.print comma;

    if i = 1 then

        printer.print quote; blank;
        printer.print "EXCLUSION      ";
        printer.print quote; comma;

    else

        printer.print quote; blank;
        printer.print "REGION_"; format$(i, "00"); "      ";
        printer.print quote; comma;

    end if

    printer.print using "#.###"; econ_data(i).region_frmfrc;
    printer.print comma;

    printer.print using " .###"; econ_data(i).region_dpf;
    printer.print comma;

    printer.print using "#####.##"; econ_data(i).region_asfp;
    printer.print comma;

    printer.print using "#####.##"; econ_data(i).region_vfrm;
    printer.print comma;

    printer.print using "#####.##"; econ_data(i).region_vnfrm

next i

exit sub

'Error return.

print_maccs_error:
    'If there is a problem, tell the user and exit function.

    msgbox error$(err)

exit sub

end sub

```

**FILENAME: OUTPFORM.FRM**

```
Version 1.00
BEGIN Form frmOutput
    AutoRedraw      = 0
    BackColor       = QBColor(3)
    BorderStyle     = 1
    Caption         = "SECPOP90 MACCS Site Data File"
    ControlBox      = -1
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(21)
    Left            = Char(1)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag             = ""
    Top             = Char(1)
    Visible         = -1
    Width           = Char(76)
    WindowState     = 0
    BEGIN TextBox txtOutput
        BackColor      = QBColor(3)
        BorderStyle    = 1
        DragMode       = 0
        Enabled        = -1
        ForeColor      = QBColor(0)
        Height          = Char(16)
        Left            = Char(0)
        MousePointer   = 0
        MultiLine       = -1
        ScrollBars     = 3
        TabIndex        = 1
        TabStop         = -1
        Tag             = ""
        Text            = ""
        Top             = Char(0)
        Visible         = -1
        Width           = Char(74)
    END
    BEGIN CommandButton cmdClose
        BackColor      = QBColor(3)
        Cancel          = -1
        Caption         = "Close"
        Default         = -1
        DragMode        = 0
        Enabled         = -1
        Height          = Char(3)
        Left            = Char(32)
        MousePointer   = 0
        TabIndex        = 0
        TabStop         = -1
        Tag             = ""
        Top             = Char(16)
        Visible         = -1
        Width           = Char(9)
    END
END

OPTION EXPLICIT

'Output (in maccs format) is displayed on this form.
```

```
'it is basically just a text viewer with a close box
'attatched to allow the user to leave.

SUB cmdClose_Click ()
    'This command allows the user to close the output
    'text box.

    frmOutput.HIDE

END SUB
```

**FILENAME: PROBCODE.BAS**

```
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

'This module contains procedures for dealing with
'verious problem file related items, such as
'loading and saving problems, and creating new problems.

SUB browse_site_file ()
    'this procedure allows the user to view and select previously created
    'site files when creating a new problem.

    DIM ForeColor AS INTEGER, BackColor AS INTEGER
    DIM Flags AS INTEGER, Cancel AS INTEGER
    DIM filename AS STRING
    STATIC Pathname AS STRING
    DIM DefaulText AS STRING, DialogTitle AS STRING

    'Initialize file dialogue box.

    DefaulText = "*.SIT"
    DialogTitle = "Browse Site"
    BackColor = WHITE
    ForeColor = BLACK
    Pathname = frmSetup.txtSite_Path.text

    'Display file choices to user, and let them select one.

    CALL FileOpen(filename, Pathname, DefaulText, DialogTitle, ForeColor,
    BackColor, Flags, Cancel)

    'If user did not select cancel from file dialogue box, then proceed.

    IF NOT Cancel THEN

        'If file is not in current directory, append pathname.

        filename = Pathname + "\\" + filename

        'Put site file name into text box on problem form.

        frmProblem_data.txtSite_file_name.text = filename

        'Mark file as modified.

        frmMain.mnuSave_ProblemCom.enabled = TRUE
        frmMain.mnuSave_As_ProblemCom.enabled = TRUE
        IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
            frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
        END IF

        'Call procedure to open site file and read in values.

        open_site_from_problem (filename)

        'If cancel was selected in file dialogue box, reset focus to
        'site file name.

    ELSE
```

```

        frmProblem_data.txtSite_file_name.SETFOCUS
    END IF

END SUB

SUB default_regions ()
    DIM h AS INTEGER, i AS INTEGER, j AS INTEGER, k AS INTEGER
    DIM m AS INTEGER, n AS INTEGER
    DIM num_sectors AS INTEGER
    DIM dividend AS INTEGER, remainder AS INTEGER

    num_sectors = (((number_of_radii - 1) * number_of_segments) + 1)
    k = 2
    frmProblem_data.lblNumber_of_regions.tag = STR$(number_of_radii)

    FOR i = 0 TO (number_of_segments)
        region_index(i, 1) = 1
    NEXT i

    IF (num_sectors <= max_econ_regions) THEN
        number_econ_regions = num_sectors

        FOR i = 0 TO (number_of_segments)
            FOR j = 2 TO (number_of_radii)
                region_index(j, i) = k
                k = k + 1
            NEXT j
        NEXT i
    ELSE
        remainder = FIX((number_of_radii - 1) MOD 6)
        IF remainder = 0 THEN
            dividend = FIX((number_of_radii - 1) / 6)
        ELSE remainder = FIX((number_of_radii - 1) MOD 5)
            dividend = FIX((number_of_radii - 1) / 5)
        END IF

        FOR h = 0 TO (number_of_segments)
            m = 2
            IF (remainder > dividend) THEN
                FOR m = 2 TO (remainder + 1)
                    region_index(m, h) = k
                NEXT m
                k = k + 1
            END IF

            FOR i = 1 TO 5
                FOR j = m TO (m + dividend - 1)
                    region_index(j, h) = k
                NEXT j
            m = j
            k = k + 1
        NEXT i

        IF remainder <= dividend THEN
            FOR i = m TO (number_of_radii)
                region_index(i, h) = k
            NEXT i
            k = k + 1
        END IF
    NEXT h
    number_econ_regions = 97

```

```

    END IF

    frmProblem_data.lblNumber_of_regions.caption = STR$(number_econ_regions)

END SUB

SUB new_problem ()

    'This procedure initializes a new problem to default settings
    'and opens the problem form.

    DIM i AS INTEGER, j AS INTEGER

    'Initialize settings to default.

    frmProblem_data.lblNumber_of_regions.caption = ""
    FOR i = 0 TO number_of_segments - 1
        FOR j = 2 TO max_number_of_radii
            region_index(j, i) = 0
        NEXT j
    NEXT i

    frmMain.mnuSave_ProblemCom.enabled = FALSE
    frmMain.mnuSave_As_ProblemCom.enabled = FALSE
    frmProblem_data.caption = CURDIR$ + "\" + "NEW_PROB.PRB"
    frmProblem_data.txtSite_file_name.text = (frmSetup.txtSite_Path.text +
    "\*.SIT")
    frmProblem_data.vsbRadius_Number.value = 1
    frmProblem_data.txtRadial_Distance1.text = ""
    frmProblem_data.txtRadial_Distance2.text = ""
    frmProblem_data.txtRadial_Distance3.text = ""
    frmProblem_data.txtPopulation_multiplier.text = "1.0"
    frmProblem_data.txtProblem_Remarks.text = ""

    FOR i = 0 TO max_number_of_radii STEP 1
        radial_distance(i) = 0
    NEXT i

    number_of_radii = 0
    number_econ_regions = 0
    x_position = 0
    y_position = 0

    'Show new problem form

    frmProblem_data.SHOW MODAL

END SUB

SUB open_problem ()

    'Open a previously created problem from file.

    DIM site_file_name AS STRING, longitude AS STRING, latitude AS STRING,
    units_of_measure AS STRING
        DIM site_name AS STRING, site_remarks AS STRING, problem_remarks AS STRING,
    dummy AS STRING
        DIM longitude_degrees AS INTEGER, longitude_minutes AS INTEGER,
    longitude_seconds AS INTEGER
        DIM latitude_degrees AS INTEGER, latitude_minutes AS INTEGER, latitude_seconds
    AS INTEGER
    DIM i AS INTEGER, j AS INTEGER
    DIM ForeColor AS INTEGER, BackColor AS INTEGER

```

```

DIM Flags AS INTEGER, Cancel AS INTEGER
DIM filename AS STRING
STATIC Pathname AS STRING
DIM DefaulText AS STRING, DialogTitle AS STRING

'Define as constants the file handle numbers for problem and site files,
'and the file buffer size.

CONST problem_file = 10, site_file = 20, file_buffer = 1024

'Initialize file dialog box.

DefaulText = "*.PRB"
DialogTitle = "Problem Open"
Pathname = frmSetup.txtProblem_path.text
BackColor = WHITE
ForeColor = BLACK

'Enable local error checking for file errors.

ON LOCAL ERROR GOTO open_prob_error

'Open file dialog box.

CALL FileOpen(filename, Pathname, DefaulText, DialogTitle, ForeColor,
BackColor, Flags, Cancel)

'If user does not select cancel from file dialogue box.

IF NOT Cancel THEN

    'if filename is not in current directory, add path.

    IF Pathname <> "" THEN filename = Pathname + "\\" + filename

    'Reset the table position variables.

    x_position = 0
    y_position = 0

    'Open filename and read problem information into temporary variables.

    OPEN filename FOR INPUT ACCESS READ LOCK WRITE AS #problem_file LEN =
file_buffer
        LINE INPUT #problem_file, dummy
        LINE INPUT #problem_file, dummy
        LINE INPUT #problem_file, site_file_name
        LINE INPUT #problem_file, dummy
        INPUT #problem_file, number_of_radii
        LINE INPUT #problem_file, dummy
        LINE INPUT #problem_file, units_of_measure
        LINE INPUT #problem_file, dummy
        FOR i = 1 TO number_of_radii STEP 1
            INPUT #problem_file, radial_distance(i)
        NEXT i
        FOR i = number_of_radii + 1 TO max_number_of_radii STEP 1
            radial_distance(i) = 0#
        NEXT i
        LINE INPUT #problem_file, dummy
        INPUT #problem_file, Population_multiplier
        LINE INPUT #problem_file, dummy
        INPUT #problem_file, number_econ_regions
        LINE INPUT #problem_file, dummy

```

```

FOR i = 0 TO number_of_segments - 1
    FOR j = 1 TO number_of_radii
        INPUT #problem_file, region_index(j, i)
    NEXT j
NEXT i
LINE INPUT #problem_file, dummy
LINE INPUT #problem_file, problem_remarks
CLOSE problem_file

'Transfer information from temporary variables to correct
'form locations.

frmProblem_data.caption = filename
frmProblem_data.txtSite_file_name.text = site_file_name
IF units_of_measure = "Kilometers" THEN
    frmProblem_data.optKilometers.value = TRUE
    frmProblem_data.optKilometers.tag = "Previously_True"
ELSE
    frmProblem_data.optmiles.value = TRUE
    frmProblem_data.optmiles.tag = "Previously_True"
END IF
frmProblem_data.vsbRadius_Number.value = 1
frmProblem_data.txtRadial_Distance1.text = RIGHT$( " " +
FORMAT$(radial_distance(1), "0.0000;0.0000;      "), 9)
frmProblem_data.txtRadial_Distance2.text = RIGHT$( " " +
FORMAT$(radial_distance(2), "0.0000;0.0000;      "), 9)
frmProblem_data.txtRadial_Distance3.text = RIGHT$( " " +
FORMAT$(radial_distance(3), "0.0000;0.0000;      "), 9)
frmProblem_data.txtPopulation_multiplier.text = RIGHT$( " " +
FORMAT$(Population_multiplier, "0.0000;0.0000;      "), 9)
frmProblem_data.txtProblem_Remarks.text = problem_remarks

'If a site file is listed, call function to read in site data.

IF site_file_name <> "" THEN
    open_site_from_problem (site_file_name)
END IF

'Set the number of economic regions to new value, and re-paint
'data form.

frmProblem_data.lblNumber_of_regions.tag = STR$(number_of_radii)
frmProblem_data.lblNumber_of_regions.caption = STR$(number_econ_regions)
CALL paint_region(0, 0)

'If problem data form is not visible, show it.

IF (frmProblem_data.visible = FALSE) THEN
    frmProblem_data.SHOW MODAL
END IF

END IF

EXIT SUB

open_prob_error:
    'On error, inform the user, and exit subroutine.

    MSGBOX ERROR$(ERR), MB_OK, "Error"
    'MSGBOX "File error - check spelling of file names."
    EXIT SUB

```

```

END SUB

SUB open_site_from_problem (file_name AS STRING)
    'Open site from problem and fill site form with data from file.

    DIM site_name AS STRING, site_remarks AS STRING, dummy AS STRING
    DIM longitude_degrees AS INTEGER, longitude_minutes AS INTEGER,
    longitude_seconds AS INTEGER
    DIM latitude_degrees AS INTEGER, latitude_minutes AS INTEGER, latitude_seconds
    AS INTEGER

    'Declare as constant the file handle number for site file and the
    'file buffer size.

    CONST site_file = 20, file_buffer = 1024

    'Enable local error checking.

    ON LOCAL ERROR GOTO open_site_problem_file_error

    'Open site file named on problem form, and read input into
    'temporary variables.

    OPEN file_name FOR INPUT ACCESS READ LOCK WRITE AS #site_file LEN =
    file_buffer
    LINE INPUT #site_file, dummy
    LINE INPUT #site_file, site_name
    LINE INPUT #site_file, dummy
    LINE INPUT #site_file, dummy
    INPUT #site_file, longitude_degrees
    INPUT #site_file, longitude_minutes
    INPUT #site_file, longitude_seconds
    LINE INPUT #site_file, dummy
    INPUT #site_file, latitude_degrees
    INPUT #site_file, latitude_minutes
    INPUT #site_file, latitude_seconds
    LINE INPUT #site_file, dummy
    LINE INPUT #site_file, site_remarks
    CLOSE site_file

    'Transfer values from temp variables to site form.

    frmSite_Data.caption = file_name
    frmSite_Data.txtSite_Name.text = site_name
    frmSite_Data.txtLongitude_Degrees.text = STR$(longitude_degrees)
    frmSite_Data.txtLongitude_Minutes.text = STR$(longitude_minutes)
    frmSite_Data.txtLongitude_Seconds.text = STR$(longitude_seconds)
    frmSite_Data.txtLatitude_Degrees.text = STR$(latitude_degrees)
    frmSite_Data.txtLatitude_Minutes.text = STR$(latitude_minutes)
    frmSite_Data.txtLatitude_Seconds.text = STR$(latitude_seconds)
    frmSite_Data.txtSite_Remarks.text = site_remarks

    EXIT SUB

open_site_problem_file_error:
    'If there is a problem, tell the user, and return to the form where
    'a correction can be made.

    IF frmProblem_data.visible = FALSE THEN
        frmProblem_data.SHOW MODELESS
        MSGBOX "Error: Unable to Open Site File."

```

```

        frmProblem_data.txtSite_file_name.SETFOCUS
        frmProblem_data.HIDE
    ELSE
        MSGBOX "Unable to Open Site File.", MB_OK, "File Error"
        frmProblem_data.txtSite_file_name.SETFOCUS
    END IF
    EXIT SUB

END SUB

SUB paint_region (x_offset AS INTEGER, y_offset AS INTEGER)

    DIM i AS INTEGER, j AS INTEGER, k AS INTEGER, x AS INTEGER, y AS INTEGER

    frmRegion.lblnum_regions.caption = STR$(number_econ_regions)
    frmRegion.lblnum_radii.caption = STR$(number_of_radii)

    IF frmProblem_data.optmiles.value = TRUE THEN
        frmRegion.lblunits.caption = "Miles"
    ELSE frmRegion.lblunits.caption = "Kilometers"
    END IF

    x_position = x_position + x_offset
    y_position = y_position + y_offset

    IF (number_of_radii < 8) THEN
        x_position = 0
        IF (x_offset <> 0) THEN
            EXIT SUB
        END IF
    ELSEIF (x_position < 0) THEN
        x_position = 0
        EXIT SUB
    ELSEIF ((x_position + 7) > number_of_radii) THEN
        x_position = x_position - 1
        EXIT SUB
    END IF

    FOR i = 1 TO 7
        x = x_position + i
        IF radial_distance(x) = 0 THEN
            frmRegion.lblRdist(i - 1).caption = ""
        ELSE frmRegion.lblRdist(i - 1).caption = LEFT$(STR$(radial_distance(x)),
4)
        END IF
    NEXT i

    k = 0

    FOR i = 0 TO 6
        IF (i = 0) AND ((y_position = 16) OR (y_position = -16)) THEN
            y_position = 0
        END IF
        IF (((i + y_position) > 15) AND (y_position > 9)) THEN
            y = (i + y_position - 16)
        ELSEIF ((y_position < 0) AND ((y_position + i) < 0)) THEN
            y = (i + y_position + 16)
        ELSE
            y = (i + y_position)
        END IF
        frmRegion.lblSector(i).caption = directions(y + 1)

        FOR j = 1 TO 7

```

```

        x = j + x_position
        IF ((radial_distance(x) = 0) OR (region_index(x, y) = 0)) THEN
            frmRegion.txtSector(k).text = ""
            frmRegion.txtSector(k).enabled = TRUE
        ELSEIF (x = 1) THEN
            frmRegion.txtSector(k).enabled = FALSE
            frmRegion.txtSector(k).text = right$( " " +
STR$(region_index(x, y)), 6)
        ELSE frmRegion.txtSector(k).text = right$( " " +
STR$(region_index(x, y)), 6)
            frmRegion.txtSector(k).enabled = TRUE
        END IF
        k = k + 1
    NEXT j
NEXT i

END SUB

SUB print_problem ()

    'Print the data on a problem form.

    DIM ForeColor AS INTEGER, BackColor AS INTEGER
    DIM Copies AS INTEGER, Cancel AS INTEGER, i AS INTEGER, j AS INTEGER
    DIM file_name AS STRING, date AS STRING, latitude AS STRING, longitude AS
STRING, site_file_name AS STRING, problem_remarks AS STRING

    'Define as constants the margins for printing.

    CONST left_margin = 10
    CONST text_width = 40

    BackColor = WHITE
    ForeColor = BLACK

    'Enable local error checking.

    ON LOCAL ERROR GOTO problem_print_error

    'Call printer dialogue box.

    CALL FilePrint(Copies, ForeColor, BackColor, Cancel)

    'If cancel is not selected from the printer dialogue box then proceed.

    IF NOT Cancel THEN

        'Execute once for each copy desired by the user.

        FOR i = 1 TO Copies

            'Send data to printer.

            printer.PRINT TAB(left_margin); "File Name:";
            file_name = frmProblem_data.caption
            CALL print_text(file_name, text_width, left_margin +
LEN("Population Multiplier: "))
            printer.PRINT
            printer.PRINT TAB(left_margin); "Site File Name:";
            site_file_name = frmProblem_data.txtSite_file_name.text
            CALL print_text(site_file_name, text_width, left_margin +
LEN("Population Multiplier: "))
            printer.PRINT

```

```

        printer.PRINT TAB(left_margin); "Number of Radii:";
        printer.PRINT USING "##"; TAB(left_margin + LEN("Population
Multiplier: ")); number_of_radii
        printer.PRINT
        printer.PRINT TAB(left_margin); "Radial Distances:";
        FOR j = 1 TO number_of_radii STEP 1
            printer.PRINT USING "#####.#####"; TAB(left_margin +
LEN("Population Multiplier: ")); radial_distance(j)
        NEXT j
        printer.PRINT
        printer.PRINT TAB(left_margin); "Population Multiplier Factor:";
        printer.PRINT USING "#####.#####"; TAB(left_margin + LEN("Population
Multiplier: ")); VAL(frmProblem_data.txtPopulation_multiplier.text)
        printer.PRINT
        printer.PRINT TAB(left_margin); "Problem Remarks:";
        problem_remarks = frmProblem_data.txtProblem_Remarks.text
        CALL print_text(problem_remarks, text_width, left_margin +
LEN("Population Multiplier: "))
        printer.NEWPAGE
    NEXT i

    'Tell printer that it is at the end of the document.
    printer.ENDDOC

    END IF

    EXIT SUB

problem_print_error:
    'If an error occurs, the user is notified, and the sub routine exited.

    MSGBOX "Device error: Check printer or other print device."
    EXIT SUB

END SUB

SUB save_as_problem ()
    'Allow the user to specify a new name for a file that is being saved.

    DIM problem_name AS STRING, problem_remarks AS STRING, dummy AS STRING,
caption AS STRING
    DIM longitude_degrees AS INTEGER, longitude_minutes AS INTEGER,
longitude_seconds AS INTEGER
    DIM latitude_degrees AS INTEGER, latitude_minutes AS INTEGER, latitude_seconds
AS INTEGER
    DIM ForeColor AS INTEGER, BackColor AS INTEGER
    DIM Flags AS INTEGER, Cancel AS INTEGER
    DIM filename AS STRING, Pathname AS STRING
    DIM DefaultText AS STRING, DialogTitle AS STRING
    DIM temp AS INTEGER
    CONST problem_file = 10, file_buffer = 1024

    'Verify that the user has entered valid input into all necessary fields
    'Do not save until all input is correct.

    temp = verify_input(1)

    IF temp = 0 THEN
        frmProblem_data.txtSite_file_name.SETFOCUS
        EXIT SUB
    ELSEIF temp = -1 THEN

```

```

        frmProblem_data.txtPopulation_multiplier.SETFOCUS
        EXIT SUB
    ELSEIF temp = -2 THEN
        frmProblem_data.txtRadial_Distance1.SETFOCUS
        EXIT SUB
    ELSEIF temp = -3 THEN
        frmProblem_data.txtRadial_Distance2.SETFOCUS
        EXIT SUB
    ELSEIF temp = -4 THEN
        MSGBOX "Setting Economic regions to default values"
        CALL default_regions
    END IF

        'Break filename into two pieces - path and filename.

        caption = frmProblem_data.caption
        CALL parse_path_and_file(caption, Pathname, filename)

        'Initialize file dialogue box.

        DefaulText = "*.PRB"
        Pathname = frmSetup.txtProblem_path.text
        DialogTitle = "Save As Problem"
        BackColor = WHITE
        ForeColor = BLACK

        'Open file dialogue box.

        CALL FileSave(filename, Pathname, DefaulText, DialogTitle, ForeColor,
        BackColor, Flags, Cancel)

        'If the user did not select cancel in the file dialogue box
        'then proceed.

        IF NOT Cancel THEN

            'If the file is not in the current directory, add the path to it.

            IF Pathname <> "" THEN filename = Pathname + "\\" + filename
            frmProblem_data.caption = filename

            'Call the function to save the problem.

            CALL save_problem

        END IF

    END SUB

SUB save_problem ()

    'Save problem data into previously named file.

    DIM caption AS STRING, i AS INTEGER, j AS INTEGER
    DIM filename AS STRING, Pathname AS STRING, tempstring AS STRING
    DIM temp AS INTEGER, filenum AS INTEGER
    CONST problem_file = 10, file_buffer = 1024

    'Remove the path from the filename.

    caption = frmProblem_data.caption
    CALL parse_path_and_file(caption, Pathname, filename)

```

```

'If it is a never-before-named problem, then call save as.

IF filename = "NEW_PROB.PRB" THEN
    CALL save_as_problem
ELSE
    'Verify that all of the data inputed by the user is correct.  If not,
    'go back and demand a correction.

    temp = verify_input(1)

    IF temp = 0 THEN
        frmProblem_data.txtSite_file_name.SETFOCUS
        EXIT SUB
    ELSEIF temp = -1 THEN
        frmProblem_data.txtPopulation_multiplier.SETFOCUS
        EXIT SUB
    ELSEIF temp = -2 THEN
        frmProblem_data.txtRadial_Distance1.SETFOCUS
        EXIT SUB
    ELSEIF temp = -3 THEN
        frmProblem_data.txtRadial_Distance2.SETFOCUS
        EXIT SUB
    ELSEIF temp = -4 THEN
        MSGBOX "Setting Economic regions to default values"
        CALL default_regions
    END IF

    IF Pathname <> "" THEN
        filename = Pathname + "\\" + filename
    END IF

    'Enable local error correction.

    ON LOCAL ERROR GOTO prob_save_error

    'Open output file, and write information to it.

    OPEN filename FOR OUTPUT ACCESS WRITE LOCK READ WRITE AS #problem_file
LEN = file_buffer
    PRINT #problem_file, "SECPOP90 V2.3"
    PRINT #problem_file, "Site File: "
    PRINT #problem_file, frmProblem_data.txtSite_file_name.text
    PRINT #problem_file, "Number of Radii:"
    PRINT #problem_file, USING "##"; number_of_radii
    PRINT #problem_file, "Units of Measure:"
    IF frmProblem_data.optKilometers.value = TRUE THEN
        PRINT #problem_file, "Kilometers"
    ELSE PRINT #problem_file, "Miles"
    END IF
    PRINT #problem_file, "Radial Distances:"
    FOR i = 1 TO number_of_radii STEP 1
        PRINT #problem_file, USING "####.####"; radial_distance(i)
    NEXT i
    PRINT #problem_file, "Population Multiplier:"
    PRINT #problem_file, USING "####.####";
VAL(frmProblem_data.txtPopulation_multiplier.text)
    PRINT #problem_file, "Number of Economic regions:"
    PRINT #problem_file, USING "##"; number_econ_regions
    PRINT #problem_file, "Economic regions:"
    FOR i = 0 TO (number_of_segments - 1)

```

```

        FOR j = 1 TO number_of_radii
            PRINT #problem_file, USING "##"; region_index(j, 1);
            PRINT #problem_file, " ";
        NEXT j
        PRINT #problem_file, ""
NEXT i
PRINT #problem_file, "Problem Remarks:"
PRINT #problem_file, frmProblem_data.txtProblem_Remarks.text
CLOSE problem_file

'Rename the problem window to the filename.

frmProblem_data.caption = filename

END IF

'If problem form is visible, set the focus to close.

IF frmProblem_data.visible = TRUE THEN frmProblem_data.cmdClose.SETFOCUS
EXIT SUB

prob_save_error:

'If an error occurs, tell the user and exit the subroutine.

MSGBOX "A FILE ERROR HAS OCCURRED. PLEASE CHECK ALL FILE NAMES"
EXIT SUB

END SUB

```

**FILENAME: PROBFORM.FRM**

```
Version 1.00
BEGIN Form frmProblem_Data
    AutoRedraw      = 0
    BackColor       = QBColor(3)
    BorderStyle     = 1
    Caption         = ""
    ControlBox      = 0
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(21)
    Left             = Char(0)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag              = ""
    Top              = Char(1)
    Visible         = -1
    Width            = Char(78)
    WindowState     = 0
    BEGIN Label Label2
        Alignment      = 0
        AutoSize        = 0
        BackColor       = QBColor(3)
        BorderStyle     = 0
        Caption         = "Site File &Name"
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(1)
        Left             = Char(3)
        MousePointer    = 0
        TabIndex        = 0
        Tag              = ""
        Top              = Char(0)
        Visible         = -1
        Width            = Char(15)
    END
    BEGIN TextBox txtSite_File_Name
        BackColor       = QBColor(3)
        BorderStyle     = 1
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(3)
        Left             = Char(2)
        MousePointer    = 0
        MultiLine       = -1
        ScrollBars      = 0
        TabIndex        = 1
        TabStop         = -1
        Tag              = ""
        Text             = ""
        Top              = Char(1)
        Visible         = -1
        Width            = Char(61)
    END
    BEGIN CommandButton cmdBrowse
        BackColor       = QBColor(3)
        Cancel          = 0
        Caption         = "S&ites"
    END
```

```

Default      = 0
DragMode     = 0
Enabled      = -1
Height       = Char(3)
Left         = Char(65)
MousePointer = 0
TabIndex     = 2
TabStop      = -1
Tag          = "Browse"
Top          = Char(1)
Visible      = -1
Width        = Char(9)

END
BEGIN Label Label7
    Alignment    = 0
    AutoSize     = 0
    BackColor    = QBColor(3)
    BorderStyle  = 0
    Caption      = "Pro&blem Remarks"
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(1)
    Left         = Char(3)
    MousePointer = 0
   TabIndex     = 14
    Tag          = ""
    Top          = Char(12)
    Visible      = -1
    Width        = Char(16)
END
BEGIN TextBox txtProblem_Remarks
    BackColor    = QBColor(3)
    BorderStyle  = 1
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(3)
    Left         = Char(2)
    MousePointer = 0
    MultiLine    = -1
    ScrollBars   = 0
   TabIndex     = 15
    TabStop      = -1
    Tag          = ""
    Text          = ""
    Top          = Char(13)
    Visible      = -1
    Width        = Char(72)
END
BEGIN CommandButton cmdOpen
    BackColor    = QBColor(3)
    Cancel       = 0
    Caption      = "P&roblems"
    Default      = 0
    DragMode     = 0
    Enabled      = -1
    Height       = Char(3)
    Left         = Char(2)
    MousePointer = 0
   TabIndex     = 16
    TabStop      = -1
    Tag          = "Open"

```

```

        Top          = Char(16)
        Visible     = -1
        Width       = Char(11)
    END
    BEGIN CommandButton cmdSave
        BackColor   = QBColor(3)
        Cancel      = 0
        Caption     = "&Save"
        Default     = 0
        DragMode    = 0
        Enabled     = -1
        Height      = Char(3)
        Left        = Char(32)
        MousePointer = 0
        TabIndex    = 18
        TabStop     = -1
        Tag         = ""
        Top         = Char(16)
        Visible     = -1
        Width       = Char(11)
    END
    BEGIN CommandButton cmdSave_As
        BackColor   = QBColor(3)
        Cancel      = 0
        Caption     = "Save &As"
        Default     = 0
        DragMode    = 0
        Enabled     = -1
        Height      = Char(3)
        Left        = Char(47)
        MousePointer = 0
        TabIndex    = 19
        TabStop     = -1
        Tag         = ""
        Top         = Char(16)
        Visible     = -1
        Width       = Char(11)
    END
    BEGIN CommandButton cmdPrint
        BackColor   = QBColor(3)
        Cancel      = 0
        Caption     = "&Print"
        Default     = 0
        DragMode    = 0
        Enabled     = -1
        Height      = Char(3)
        Left        = Char(62)
        MousePointer = 0
        TabIndex    = 20
        TabStop     = -1
        Tag         = ""
        Top         = Char(16)
        Visible     = -1
        Width       = Char(11)
    END
    BEGIN Label Label5
        Alignment   = 2
        AutoSize    = 0
        BackColor   = QBColor(3)
        BorderStyle = 0
        Caption     = "Radial &Distances"
        DragMode    = 0
        Enabled     = -1

```

```

ForeColor      = QBColor(0)
Height        = Char(2)
Left          = Char(30)
MousePointer  = 0
TabIndex      = 3
Tag           = ""
Top           = Char(5)
Visible       = -1
Width         = Char(9)
END
BEGIN Label Label1
    Alignment    = 2
    AutoSize     = 0
    BackColor    = QBColor(3)
    BorderStyle  = 0
    Caption      = "Radial Units"
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(2)
    Left         = Char(43)
    MousePointer = 0
   TabIndex     = 29
    Tag          = ""
    Top          = Char(5)
    Visible      = -1
    Width        = Char(6)
END
BEGIN Label lblRegions
    Alignment    = 2
    AutoSize     = 0
    BackColor    = QBColor(3)
    BorderStyle  = 0
    Caption      = "Number of Economic Regions"
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(2)
    Left         = Char(58)
    MousePointer = 0
   TabIndex     = 9
    Tag          = ""
    Top          = Char(5)
    Visible      = -1
    Width        = Char(16)
END
BEGIN Frame fraRadius_Numbers
    BackColor    = QBColor(3)
    Caption      = ""
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(5)
    Left         = Char(23)
    MousePointer = 0
   TabIndex     = 22
    Tag          = ""
    Top          = Char(7)
    Visible      = -1
    Width        = Char(4)
    BEGIN Label lblRadius1
        Alignment    = 0
        AutoSize     = 0

```

```

        BackColor      = QBColor(3)
        BorderStyle   = 0
        Caption       = " 1"
        DragMode      = 0
        Enabled       = -1
        ForeColor     = QBColor(0)
        Height        = Char(1)
        Left          = Char(0)
        MousePointer = 0
        TabIndex      = 23
        Tag           = ""
        Top           = Char(0)
        Visible       = -1
        Width         = Char(2)
    END
    BEGIN Label lblRadius3
        Alignment     = 0
        AutoSize      = 0
        BackColor     = QBColor(3)
        BorderStyle   = 0
        Caption       = " 3"
        DragMode      = 0
        Enabled       = -1
        ForeColor     = QBColor(0)
        Height        = Char(1)
        Left          = Char(0)
        MousePointer = 0
        TabIndex      = 25
        Tag           = ""
        Top           = Char(2)
        Visible       = -1
        Width         = Char(2)
    END
    BEGIN Label lblRadius2
        Alignment     = 0
        AutoSize      = 0
        BackColor     = QBColor(3)
        BorderStyle   = 0
        Caption       = " 2"
        DragMode      = 0
        Enabled       = -1
        ForeColor     = QBColor(0)
        Height        = Char(1)
        Left          = Char(0)
        MousePointer = 0
        TabIndex      = 24
        Tag           = ""
        Top           = Char(1)
        Visible       = -1
        Width         = Char(2)
    END
    END
    BEGIN Frame fraRadial_Distances
        BackColor     = QBColor(3)
        Caption       = ""
        DragMode      = 0
        Enabled       = -1
        ForeColor     = QBColor(0)
        Height        = Char(5)
        Left          = Char(28)
        MousePointer = 0
        TabIndex      = 26
        Tag           = ""
    END

```

```

        Top          = Char(7)
        Visible      = -1
        Width        = Char(12)
BEGIN TextBox txtRadial_Distance1
        BackColor    = QBColor(3)
        BorderStyle  = 0
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(1)
        Left         = Char(0)
        MousePointer = 0
        MultiLine    = -1
        ScrollBars   = 0
        TabIndex     = 4
        TabStop      = -1
        Tag          = ""
        Text         = ""
        Top          = Char(0)
        Visible      = -1
        Width        = Char(10)
END
BEGIN TextBox txtRadial_Distance2
        BackColor    = QBColor(3)
        BorderStyle  = 0
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(1)
        Left         = Char(0)
        MousePointer = 0
        MultiLine    = -1
        ScrollBars   = 0
        TabIndex     = 5
        TabStop      = -1
        Tag          = ""
        Text         = ""
        Top          = Char(1)
        Visible      = -1
        Width        = Char(10)
END
BEGIN TextBox txtRadial_Distance3
        BackColor    = QBColor(3)
        BorderStyle  = 0
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(1)
        Left         = Char(0)
        MousePointer = 0
        MultiLine    = -1
        ScrollBars   = 0
        TabIndex     = 6
        TabStop      = -1
        Tag          = ""
        Text         = ""
        Top          = Char(2)
        Visible      = -1
        Width        = Char(10)
END
END
BEGIN Frame Frame1
        BackColor    = QBColor(3)

```

```

Caption      = ""
DragMode     = 0
Enabled      = -1
ForeColor    = QBColor(0)
Height       = Char(4)
Left         = Char(41)
MousePointer = 0
TabIndex     = 28
Tag          = ""
Top          = Char(7)
Visible      = -1
Width        = Char(10)
BEGIN OptionButton optKilometers
    BackColor   = QBColor(3)
    Caption     = "&KM"
    DragMode    = 0
    Enabled     = -1
    ForeColor   = QBColor(0)
    Height      = Char(1)
    Left        = Char(1)
    MousePointer = 0
   TabIndex     = 7
    TabStop     = 0
    Tag         = ""
    Top         = Char(0)
    Value       = 0
    Visible     = -1
    Width       = Char(7)
END
BEGIN OptionButton optMiles
    BackColor   = QBColor(3)
    Caption     = "&MI"
    DragMode    = 0
    Enabled     = -1
    ForeColor   = QBColor(0)
    Height      = Char(1)
    Left        = Char(1)
    MousePointer = 0
   TabIndex     = 8
    TabStop     = -1
    Tag         = ""
    Top         = Char(1)
    Value       = -1
    Visible     = -1
    Width       = Char(7)
END
END
BEGIN CommandButton cmdEdit_regions
    BackColor   = QBColor(3)
    Cancel      = 0
    Caption     = "&Edit Regions"
    Default     = 0
    DragMode    = 0
    Enabled     = -1
    Height      = Char(3)
    Left        = Char(58)
    MousePointer = 0
   TabIndex     = 11
    TabStop     = -1
    Tag         = ""
    Top         = Char(10)
    Visible     = -1
    Width       = Char(16)

```

```

END
BEGIN Label Label4
  Alignment      = 2
  AutoSize       = 0
  BackColor      = QBColor(3)
  BorderStyle    = 0
  Caption        = "Radius Number"
  DragMode       = 0
  Enabled        = -1
  ForeColor      = QBColor(0)
  Height         = Char(2)
  Left           = Char(22)
  MousePointer   = 0
  TabIndex       = 21
  Tag            = ""
  Top            = Char(5)
  Visible        = -1
  Width          = Char(6)
END
BEGIN Label lblNumber_of_regions
  Alignment      = 1
  AutoSize       = 0
  BackColor      = QBColor(3)
  BorderStyle    = 1
  Caption        = " "
  DragMode       = 0
  Enabled        = -1
  ForeColor      = QBColor(0)
  Height         = Char(3)
  Left           = Char(58)
  MousePointer   = 0
  TabIndex       = 10
  Tag            = ""
  Top            = Char(7)
  Visible        = -1
  Width          = Char(16)
END
BEGIN VScrollBar vsbRadius_Number
  Attached       = 0
  DragMode       = 0
  Enabled        = -1
  Height         = Char(3)
  LargeChange    = 1
  Left           = Char(39)
  Max            = 48
  Min            = 1
  MousePointer   = 0
  SmallChange    = 1
  TabIndex       = 27
  TabStop        = 0
  Tag            = ""
  Top            = Char(8)
  Value          = 1
  Visible        = -1
  Width          = Char(1)
END
BEGIN Label Label6
  Alignment      = 2
  AutoSize       = 0
  BackColor      = QBColor(3)
  BorderStyle    = 0
  Caption        = "Population Multiplier"
  DragMode       = 0

```

```

        Enabled      = -1
        ForeColor   = QBColor(0)
        Height      = Char(2)
        Left        = Char(4)
        MousePointer = 0
        TabIndex    = 12
        Tag         = ""
        Top         = Char(5)
        Visible     = -1
        Width       = Char(11)
    END
    BEGIN TextBox txtPopulation_multiplier
        BackColor   = QBColor(3)
        BorderStyle = 1
        DragMode    = 0
        Enabled     = -1
        ForeColor   = QBColor(0)
        Height      = Char(3)
        Left        = Char(3)
        MousePointer = 0
        MultiLine   = 0
        ScrollBars  = 0
        TabIndex    = 13
        TabStop     = -1
        Tag         = ""
        Text        = ""
        Top         = Char(7)
        Visible     = -1
        Width       = Char(12)
    END
    BEGIN CommandButton cmdClose
        BackColor   = QBColor(3)
        Cancel      = -1
        Caption     = "&Close"
        Default     = 0
        DragMode    = 0
        Enabled     = -1
        Height      = Char(3)
        Left        = Char(17)
        MousePointer = 0
        TabIndex    = 17
        TabStop     = -1
        Tag         = "Close"
        Top         = Char(16)
        Visible     = -1
        Width       = Char(11)
    END
END
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

'This form allows the user to create, edit, and save different problems

SUB cmdBrowse_Click ()
    'Call the function which allows the user to see which
    'site files are available.  For more info, see browse_site_file.

    CALL browse_site_file

END SUB

```

```

SUB cmdClose_Click ()
    'This routine allows the user to close the problem
    'window without saving.

    DIM temp AS INTEGER, filenum AS INTEGER
    DIM tempstring AS STRING

    'Verify that all of the data entered by the user
    'is valid.
    temp = verify_input(1)

    'Remove the problem data form from the screen.

    frmProblem_data.HIDE

END SUB

SUB cmdEdit_regions_Click ()
    IF (number_of_radii < 2) THEN
        MSGBOX "Error: At least two radii must be specified before regions are
edited."
        EXIT SUB
    ELSEIF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
        frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
    END IF

    IF (number_econ_regions = 0) OR (number_econ_regions > max_econ_regions) OR
(number_econ_regions < min_econ_regions) THEN
        CALL default_regions
    END IF

    x_position = 0
    y_position = 0

    CALL paint_region(0, 0)

    frmRegion.SHOW MODAL

END SUB

SUB cmdOpen_Click ()
    'This routine allows the user to open a previously
    'defined problem from the problem form.  For more
    'info see the open_problem procedure

    CALL open_problem

END SUB

SUB cmdPrint_Click ()
    'This routine allows the user to print the data in
    'a problem file.  For more information
    'see the print_problem procedure.

    CALL print_problem

END SUB

SUB cmdSave_As_Click ()

```

```

'This routine allows the user to save a file by a
'new name. For more information, see the
'save_as_problem procedure.

CALL save_as_problem

END SUB

SUB cmdSave_Click ()

'This routine allows the user to save data entered
'in a problem form. For more information,
'see the save_problem procedure.

CALL save_problem

END SUB

SUB Form_Unload (Cancel AS INTEGER)

'This routine allows the user to close the problem
>window without saving.

DIM temp AS INTEGER, filenum AS INTEGER
DIM tempstring AS STRING

'Verify that all of the data entered by the user
'is valid.
temp = verify_input(1)

'Remove the problem data form from the screen.

frmProblem_data.HIDE
LOAD frmProblem_data
frmProblem_data.SHOW MODAL

END SUB

SUB optKilometers_GotFocus ()

'This event procedure allows the user to convert miles to kilometers if
'necessary, and it allows the user to specify which unit measures
'should be in.

DIM i AS INTEGER, j AS INTEGER, reply AS INTEGER, tempstring AS STRING, tempchar
AS STRING

'If user had not previously specified kilometers, ask if a conversion
'is necessary.

IF (optkilometers.tag <> "Previously_True") THEN
    reply = MSGBOX("Do you wish to convert any existing radial" + CHR$(13) +
distances from Miles to Kilometers?", MB_YESNOCANCEL + MB_DEFBUTTON2, "Conversion")
    IF reply = IDYES THEN

        FOR i = 1 TO number_of_radii
            IF (radial_distance(i) < .125) AND (radial_distance(i) <> 0) THEN
                MSGBOX "Error: unable to convert to kilometers distances of less
than .125 miles."
                optmiles.value = TRUE
                optmiles.SETFOCUS
                EXIT SUB
    END IF
END IF

```

```

ELSEIF radial_distance(1) > 6213 THEN
    MSGBOX "Error: unable to convert to kilometers distances of more
than 6213 miles."
        optmiles.value = TRUE
        optmiles.SETFOCUS
        EXIT SUB
    END IF
NEXT 1

'If conversion is requested, perform the math.

FOR 1 = 1 TO number_of_radii STEP 1
    tempstring = STR$(radial_distance(1) * miles_to_kilometers)
    tempchar = "0"

    'Examine the converted number, and round it to three digits
    'after the decimal point.

    FOR j = 1 TO (LEN(tempstring) - 3)
        IF (MID$(tempstring, j, 1) = ".") THEN
            tempchar = MID$(tempstring, j + 4, 1)
            tempstring = LEFT$(tempstring, j + 3)
        END IF
    NEXT j

    radial_distance(1) = VAL(tempstring)

    IF (radial_distance(1) >= 9999) THEN
        radial_distance(1) = 9999
    ELSEIF (radial_distance(1) <= .1) THEN
        radial_distance(1) = .1
    END IF

    IF (VAL(tempchar) <= 5) THEN radial_distance(1) = radial_distance(1) +
.001
    NEXT 1

    'Change the radial distances already listed to new unit of
    'measure.

    frmProblem_data.txtRadial_distance1.text = RIGHT$("      " +
FORMAT$(radial_distance(vsbRadius_number.value), "0.0000;0.0000;      "), 9)
    frmProblem_data.txtRadial_distance2.text = RIGHT$("      " +
FORMAT$(radial_distance(vsbRadius_number.value + 1), "0.0000;0.0000;      "), 9)
    frmProblem_data.txtRadial_distance3.text = RIGHT$("      " +
FORMAT$(radial_distance(vsbRadius_number.value + 2), "0.0000;0.0000;      "), 9)

ELSEIF reply = IDNO THEN
    FOR 1 = 1 TO number_of_radii
        IF (radial_distance(1) < .1) OR (radial_distance(1) > 9999) THEN
            MSGBOX "Error: Radial distance out of bounds"
            optmiles.value = TRUE
            optmiles.tag = "Previously_True"
            optmiles.SETFOCUS
            EXIT SUB
        END IF
    NEXT 1
END IF

'If necessary, mark which unit is recognized as being the
'previous unit.

```

```

IF reply <> IDCANCEL THEN
    optkilometers.tag = "Previously_True"
    optmiles.tag = ""

    'Mark form as modified.

    frmMain.mnuSave_ProblemCom.Enabled = TRUE
    frmMain.mnuSave_As_ProblemCom.Enabled = TRUE
    IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
        frmProblem_data.caption = frmProblem_data.caption + " (Modified)"

    END IF

    ELSE
        'if cancel, then don't change anything on form.

        optmiles.value = TRUE
        optmiles.SETFOCUS

    END IF

    END IF

END SUB

SUB optMiles_GotFocus ()
    'Allow the user to convert kilometers to miles or enter data in miles.

    DIM i AS INTEGER, j AS INTEGER, reply AS INTEGER, tempstring AS STRING, tempchar
    AS STRING

    'If user has changed units then proceed.

    IF (optmiles.tag <> "Previously_True") THEN
        reply = MSGBOX("Do you wish to convert any existing radial" + CHR$(13) + "
distances from Kilometers to Miles?", MB_YESNOCANCEL + MB_DEFBUTTON2, "Conversion")

        'If user wishes to convert numbers to new units, then proceed.

        IF reply = IDYES THEN

            FOR i = 1 TO number_of_radii
                IF (radial_distance(i) <= .2) AND (radial_distance(i) <> 0) THEN
                    MSGBOX "Error: unable to convert to miles distances of less than
.2 kilometers."
                    optkilometers.value = TRUE
                    optkilometers.SETFOCUS
                    EXIT SUB
                ELSEIF radial_distance(i) > 9999 THEN
                    MSGBOX "Error: unable to convert to miles distances of more than
9999 kilometers."
                    optkilometers.value = TRUE
                    optkilometers.SETFOCUS
                    EXIT SUB
                END IF
            NEXT i

            FOR i = 1 TO number_of_radii STEP 1
                tempchar = "0"
                tempstring = STR$(radial_distance(i) * kilometers_to_miles)

```

```

'Analyze converted number and round it to three decimal
'places.

FOR j = 1 TO (LEN(tempstring) - 3)
    IF (MID$(tempstring, j, 1) = ".") THEN
        tempchar = MID$(tempstring, j + 4, 1)
        tempstring = LEFT$(tempstring, j + 3)
    END IF
NEXT j

radial_distance(1) = VAL(tempstring)

IF (radial_distance(1) >= 6212.999) THEN
    radial_distance(1) = 6213
ELSEIF (radial_distance(1) <= .07) THEN
    radial_distance(1) = .07
END IF

IF VAL(tempchar) > 4 THEN radial_distance(1) = radial_distance(1) +
.001
NEXT 1

'Write new radial distances to problem form.

frmProblem_data.txtRadial_distance1.text = RIGHT$("      " +
FORMAT$(radial_distance(vsbRadius_number.value), "0.0000;0.0000;      "), 9)
frmProblem_data.txtRadial_distance2.text = RIGHT$("      " +
FORMAT$(radial_distance(vsbRadius_number.value + 1), "0.0000;0.0000;      "), 9)
frmProblem_data.txtRadial_distance3.text = RIGHT$("      " +
FORMAT$(radial_distance(vsbRadius_number.value + 2), "0.0000;0.0000;      "), 9)

ELSEIF reply = IDNO THEN
    FOR 1 = 1 TO number_of_radii
        IF (radial_distance(1) < .07) OR (radial_distance(1) > 6213) THEN
            MSGBOX "Error: Radial distance out of bounds"
            optkilometers.value = TRUE
            optkilometers.tag = "Previously_True"
            optkilometers.SETFOCUS
            EXIT SUB
        END IF
    NEXT 1
END IF

'If user has not canceled, update which unit was previously used.

IF reply <> IDCANCEL THEN
    optmiles.tag = "Previously_True"
    optkilometers.tag = ""

'Mark form as modified.

frmMain.mnuSave_ProblemCom.Enabled = TRUE
frmMain.mnuSave_As_ProblemCom.Enabled = TRUE
IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
    frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
END IF
ELSE

'If cancel is selected, leave values as they were.

optkilometers.value = TRUE
optkilometers.SETFOCUS

```

```

    END IF

    END IF

END SUB

SUB txtPopulation_multiplier_KeyPress (KeyAscii AS INTEGER)

    'This routine marks a problem form as modified if keys are pressed in
    'the population multiplier field.

    frmMain.mnuSave_ProblemCom.Enabled = TRUE
    frmMain.mnuSave_As_ProblemCom.Enabled = TRUE

    IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
        frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
    END IF

END SUB

SUB txtProblem_Remarks_KeyPress (KeyAscii AS INTEGER)

    'This routine marks a problem as modified if keys are pressed, and
    'deals with the enter key being pressed.

    'If enter key is pressed, display message.

    IF KeyAscii = KEY_RETURN THEN

        'Erase keystroke.

        KeyAscii = 0
        MSGBOX "Sorry, no hard returns allowed." + CHR$(13) + "Words will wrap
automatically.", MB_OK, "Error"

        'Mark problem form as modified.

    ELSE

        frmMain.mnuSave_ProblemCom.Enabled = TRUE
        frmMain.mnuSave_As_ProblemCom.Enabled = TRUE

        IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
            frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
        END IF

    END IF

END SUB

SUB txtProblem_Title_KeyPress (KeyAscii AS INTEGER)

    'This routine marks the problem form as modified if a key is pressed
    'within the title field. It also deals with the user pressing enter.

    'If the user presses the enter key, display warning message.

    IF KeyAscii = KEY_RETURN THEN

        'Erase keystroke.

        KeyAscii = 0

```

```

    MSGBOX "Sorry, no hard returns allowed." + CHR$(13) + "Words will wrap
automatically.", MB_OK, "Error"

ELSE

    'Mark problem form as modified.

    frmMain.mnuSave_ProblemCom.Enabled = TRUE
    frmMain.mnuSave_As_ProblemCom.Enabled = TRUE

    IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
        frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
    END IF

END IF

END SUB

SUB txtRadial_Distance1_KeyDown (KeyCode AS INTEGER, Shift AS INTEGER)

    'This routine allows the user to scroll up and down on the list of
    'radial distances.

    'If up arrow on keyboard has been pressed then scroll up one entry.

    IF KeyCode = KEY_UP THEN

        IF vsbRadius_number.value > 1 THEN
            CALL txtRadial_Distance1_LostFocus
            vsbRadius_number.value = vsbRadius_number.value - 1
        ELSE
            CALL txtRadial_Distance1_LostFocus
        END IF

    'If down arrow has been pressed, scroll down one position.

    ELSEIF KeyCode = KEY_DOWN THEN

        txtRadial_distance2.SETFOCUS

    END IF

END SUB

SUB txtRadial_Distance1_KeyPress (KeyAscii AS INTEGER)

    'This routine marks the problem form as modified if a key is pressed
    'in the distance1 field.

    frmMain.mnuSave_ProblemCom.Enabled = TRUE
    frmMain.mnuSave_As_ProblemCom.Enabled = TRUE

    IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
        frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
    END IF

    IF KeyAscii = KEY_RETURN THEN
        txtRadial_distance2.SETFOCUS
    END IF

END SUB

SUB txtRadial_Distance1_LostFocus ()


```

```

'This routine sorts the entries in the radial distance list so that they
'appear in ascending order, and updates the list

DIM temp AS DOUBLE, units AS INTEGER
DIM zero_found AS INTEGER, radius_zero AS INTEGER
DIM radius_edited AS INTEGER, i AS INTEGER, j AS INTEGER
DIM tempchar AS STRING, tempstring AS STRING

'Record the value of the newly edited radius into the radial
'distance array.

radius_edited = VAL(lblRadius1.caption)
radial_distance(radius_edited) = VAL(txtRadial_distance1.text)

'Examine the Radius, and round it to three digits after the decimal.

tempstring = STR$(radial_distance(radius_edited))
FOR j = 1 TO (LEN(tempstring) - 3)
    IF (MID$(tempstring, j, 1) = ".") THEN
        tempchar = MID$(tempstring, j + 4, 1)
        tempstring = LEFT$(tempstring, j + 3)
    END IF
NEXT j
radial_distance(radius_edited) = VAL(tempstring)

'Check to see if the radius is within acceptable limits for maccs
'compatability.

IF (radial_distance(radius_edited) = 0) THEN
    radius_zero = TRUE
ELSEIF TYPEOF screen.activecontrol IS OptionButton THEN
ELSEIF (optmiles.value = TRUE) THEN
    IF (radial_distance(radius_edited) > 6213) THEN
        MSGBOX "Error: this program can't process radii greater than 6213
Miles."
        radial_distance(radius_edited) = 0
        txtRadial_distance1.text = ""
        radius_zero = TRUE
    ELSEIF (radial_distance(radius_edited) < .0699) THEN
        MSGBOX "Error: this program can't process radii smaller than .07 Miles."
        radial_distance(radius_edited) = 0
        txtRadial_distance1.text = ""
        radius_zero = TRUE
    END IF
ELSE
    IF (radial_distance(radius_edited) > 9999) THEN
        MSGBOX "Error: this program can't process radii greater than 9999
Kilometers."
        radial_distance(radius_edited) = 0
        txtRadial_distance1.text = ""
        radius_zero = TRUE
    ELSEIF (radial_distance(radius_edited) < .099) THEN
        MSGBOX "Error: this program can't process radii smaller than .1
Kilometers."
        radial_distance(radius_edited) = 0
        txtRadial_distance1.text = ""
        radius_zero = TRUE
    END IF
END IF

'If radius is not equal to zero, check to see if it is equal to
'any previous radial distance. If it is, set it equal to zero.

```

```

IF NOT radius_zero THEN
    FOR i = 1 TO radius_edited - 1 STEP 1
        IF radial_distance(radius_edited) = radial_distance(i) THEN
            radial_distance(radius_edited) = 0
            radius_zero = TRUE
            EXIT FOR
        END IF
    NEXT i
END IF

'If distance is still not equal to zero, check to see if it is equal
'to any following radial distance. If it is, set it equal to zero.

IF NOT radius_zero THEN
    FOR i = radius_edited + 1 TO max_number_of_radii STEP 1
        IF radial_distance(radius_edited) = radial_distance(i) THEN
            radial_distance(i) = 0
            radius_zero = TRUE
            EXIT FOR
        END IF
    NEXT i
END IF

'Use standard sorting algorithm to put radii in ascending order.

FOR i = 1 TO max_number_of_radii - 1 STEP 1
    FOR j = i + 1 TO max_number_of_radii STEP 1
        IF (radial_distance(j) <> 0) THEN
            IF ((radial_distance(i) > radial_distance(j)) OR (radial_distance(i)
= 0)) THEN
                temp = radial_distance(i)
                radial_distance(i) = radial_distance(j)
                radial_distance(j) = temp
                IF (NOT radius_zero) THEN
                    IF i = radius_edited THEN
                        radius_edited = j
                    ELSEIF j = radius_edited THEN
                        radius_edited = i
                    END IF
                END IF
            END IF
        END IF
    NEXT j
NEXT i

'Determine where list of radii ends.

FOR i = 1 TO max_number_of_radii STEP 1
    IF radial_distance(i) = 0 THEN
        EXIT FOR
    END IF
NEXT i

'Determine number of radii.

number_of_radii = i - 1

IF radius_edited > max_number_of_radii - 2 THEN
    radius_edited = max_number_of_radii - 2
END IF

'Update user display (problem form).

```

```

lblRadius1.caption = RIGHT$(STR$(radius_edited), 2)
txtRadial_distance1.text = RIGHT$("    " +
FORMAT$(radial_distance(radius_edited), "0.0000;0.0000;      "), 9)
lblRadius2.caption = RIGHT$(STR$(radius_edited + 1), 2)
txtRadial_distance2.text = RIGHT$("    " + FORMAT$(radial_distance(radius_edited
+ 1), "0.0000;0.0000;      "), 9)
lblRadius3.caption = RIGHT$(STR$(radius_edited + 2), 2)
txtRadial_distance3.text = RIGHT$("    " + FORMAT$(radial_distance(radius_edited
+ 2), "0.0000;0.0000;      "), 9)

vsbRadius_number.value = radius_edited

IF (VAL(frmProblem_data.lblNumber_of_regions.tag) <> number_of_radius) THEN
    CALL default_regions
    CALL paint_region(0, 0)
END IF

END SUB

SUB txtRadial_Distance2_KeyDown (KeyCode AS INTEGER, Shift AS INTEGER)

    'If an arrow key is pressed, then call the distance 1 function
    'to scroll radial distances up or down.

    IF KeyCode = KEY_UP THEN
        txtRadial_distance1.SETFOCUS
    ELSEIF KeyCode = KEY_DOWN THEN
        txtRadial_distance3.SETFOCUS
    END IF

END SUB

SUB txtRadial_Distance2_KeyPress (KeyAscii AS INTEGER)

    'This routine marks the problem form as modified, and goes to the next
    'radius if enter is pressed.

    frmMain.mnuSave_ProblemCom.Enabled = TRUE
    frmMain.mnuSave_As_ProblemCom.Enabled = TRUE

    IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
        frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
    END IF

    IF KeyAscii = KEY_RETURN THEN
        KeyAscii = 0
        txtRadial_distance3.SETFOCUS
    END IF

END SUB

SUB txtRadial_Distance2_LostFocus ()

    'See radial_Distance1_lostfocus for documentation of this routine.

    DIM temp AS DOUBLE
    DIM zero_found AS INTEGER, radius_zero AS INTEGER
    DIM radius_edited AS INTEGER, i AS INTEGER, j AS INTEGER
    DIM tempstring AS STRING, tempchar AS STRING

    radius_edited = VAL(lblRadius2.caption)
    radial_distance(radius_edited) = VAL(txtRadial_distance2.text)

```

```

tempstring = STR$(radial_distance(radius_edited))
FOR j = 1 TO (LEN(tempstring) - 3)
    IF (MID$(tempstring, j, 1) = ".") THEN
        tempchar = MID$(tempstring, j + 4, 1)
        tempstring = LEFT$(tempstring, j + 3)
    END IF
NEXT j
radial_distance(radius_edited) = VAL(tempstring)

IF (radial_distance(radius_edited) = 0) THEN
    radius_zero = TRUE
ELSEIF TYPEOF screen.activecontrol IS OptionButton THEN
ELSEIF (optmiles.value = TRUE) THEN
    IF (radial_distance(radius_edited) > 6213) THEN
        MSGBOX "Error: this program can't process radii greater than 6213
Miles."
        radial_distance(radius_edited) = 0
        txtRadial_distance2.text = ""
        radius_zero = TRUE
    ELSEIF (radial_distance(radius_edited) < .0695) THEN
        MSGBOX "Error: this program can't process radii smaller than .07 Miles."
        radial_distance(radius_edited) = 0
        radius_zero = TRUE
    END IF
ELSE
    IF (radial_distance(radius_edited) > 9999) THEN
        MSGBOX "Error: this program can't process radii greater than 9999
Kilometers."
        radial_distance(radius_edited) = 0
        txtRadial_distance2.text = ""
        radius_zero = TRUE
    ELSEIF (radial_distance(radius_edited) < .099) THEN
        MSGBOX "Error: this program can't process radii smaller than .1
Kilometers."
        radial_distance(radius_edited) = 0
        txtRadial_distance2.text = ""
        radius_zero = TRUE
    END IF
END IF

IF NOT radius_zero THEN
    FOR i = 1 TO radius_edited - 1 STEP 1
        IF radial_distance(radius_edited) = radial_distance(i) THEN
            radial_distance(radius_edited) = 0
            radius_zero = TRUE
            EXIT FOR
        END IF
    NEXT i
END IF

IF NOT radius_zero THEN
    FOR i = radius_edited + 1 TO max_number_of_radii STEP 1
        IF radial_distance(radius_edited) = radial_distance(i) THEN
            radial_distance(radius_edited) = 0
            radius_zero = TRUE
            EXIT FOR
        END IF
    NEXT i
END IF

FOR i = 1 TO max_number_of_radii - 1 STEP 1
    FOR j = i + 1 TO max_number_of_radii STEP 1

```

```

        IF (radial_distance(j) <> 0) THEN
            IF ((radial_distance(i) > radial_distance(j)) OR (radial_distance(i)
= 0)) THEN
                temp = radial_distance(i)
                radial_distance(i) = radial_distance(j)
                radial_distance(j) = temp
                IF (NOT radius_zero) THEN
                    IF i = radius_edited THEN
                        radius_edited = j
                    ELSEIF j = radius_edited THEN
                        radius_edited = i
                    END IF
                END IF
            END IF
        END IF
    NEXT j
NEXT i

i = 1
zero_found = FALSE

WHILE ((NOT zero_found) AND (i <= max_number_of_radii))
    IF radial_distance(i) = 0 THEN
        zero_found = TRUE
    ELSE
        i = i + 1
    END IF
WEND

number_of_radii = i - 1

IF radius_edited > max_number_of_radii - 1 THEN
    radius_edited = max_number_of_radii - 1
ELSEIF radius_edited < 2 THEN
    radius_edited = 2
END IF

lblRadius1.caption = RIGHT$(STR$(radius_edited - 1), 2)
txtRadial_distance1.text = RIGHT$("    " + FORMAT$(radial_distance(radius_edited
- 1), "0.0000;0.0000;      "), 9)
lblRadius2.caption = RIGHT$(STR$(radius_edited), 2)
txtRadial_distance2.text = RIGHT$("    " +
FORMAT$(radial_distance(radius_edited), "0.0000;0.0000;      "), 9)
lblRadius3.caption = RIGHT$(STR$(radius_edited + 1), 2)
txtRadial_distance3.text = RIGHT$("    " + FORMAT$(radial_distance(radius_edited
+ 1), "0.0000;0.0000;      "), 9)

vsbRadius_number.value = radius_edited - 1

IF (VAL(frmProblem_data.lblNumber_of_regions.tag) <> number_of_radii) THEN
    CALL default_regions
    CALL paint_region(0, 0)
END IF

END SUB

SUB txtRadial_Distance3_KeyDown (KeyCode AS INTEGER, Shift AS INTEGER)
    'See txtRadial_distance1_KeyDown for documentation of this routine.

    IF KeyCode = KEY_UP THEN
        txtRadial_distance2.SETFOCUS
    ELSEIF KeyCode = KEY_DOWN THEN

```

```

        IF vsbRadius_number.value < max_number_of_radii - 2 THEN
            CALL txtRadial_Distance3_LostFocus
            vsbRadius_number.value = vsbRadius_number.value + 1
        ELSE
            CALL txtRadial_Distance3_LostFocus
        END IF
    END IF

END SUB

SUB txtRadial_Distance3_KeyPress (KeyAscii AS INTEGER)

    'See frmTxtRadial_distance1_KeyPress for documentation of this function.

    frmMain.mnuSave_ProblemCom.Enabled = TRUE
    frmMain.mnuSave_As_ProblemCom.Enabled = TRUE

    IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
        frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
    END IF

    IF KeyAscii = KEY_RETURN THEN
        CALL txtRadial_Distance3_LostFocus
        IF vsbRadius_number.value < max_number_of_radii - 2 THEN
            vsbRadius_number.value = vsbRadius_number.value + 1
        END IF
    END IF

END SUB

SUB txtRadial_Distance3_LostFocus ()

    'See txtRadial_distance1_LostFocus for documentation of this function.

    DIM temp AS DOUBLE
    DIM zero_found AS INTEGER, radius_zero AS INTEGER
    DIM radius_edited AS INTEGER, i AS INTEGER, j AS INTEGER
    DIM tempstring AS STRING, tempchar AS STRING

    radius_edited = VAL(lblRadius3.caption)
    radial_distance(radius_edited) = VAL(txtRadial_distance3.text)

    tempstring = STR$(radial_distance(radius_edited))
    FOR j = 1 TO (LEN(tempstring) - 3)
        IF (MID$(tempstring, j, 1) = ".") THEN
            tempchar = MID$(tempstring, j + 4, 1)
            tempstring = LEFT$(tempstring, j + 3)
        END IF
    NEXT j
    radial_distance(radius_edited) = VAL(tempstring)

    IF (radial_distance(radius_edited) = 0) THEN
        radius_zero = TRUE
    ELSEIF TYPEOF screen.activecontrol IS OptionButton THEN
    ELSEIF (optmiles.value = TRUE) THEN
        IF (radial_distance(radius_edited) > 6213) THEN
            MSGBOX "Error: this program can't process radii greater than 6213
Miles."
            radial_distance(radius_edited) = 0
            radius_zero = TRUE
        ELSEIF (radial_distance(radius_edited) < .0695) THEN
            MSGBOX "Error: this program can't process radii smaller than .07 Miles."
            radial_distance(radius_edited) = 0
        END IF
    END IF
END SUB

```

```

        radius_zero = TRUE
    END IF
ELSE
    IF (radial_distance(radius_edited) > 9999) THEN
        MSGBOX "Error: this program can't process radii greater than 9999
Kilometers."
        radial_distance(radius_edited) = 0
        radius_zero = TRUE
    ELSEIF (radial_distance(radius_edited) < .099) THEN
        MSGBOX "Error: this program can't process radii smaller than .1
Kilometers."
        radial_distance(radius_edited) = 0
        radius_zero = TRUE
    END IF
END IF

IF NOT radius_zero THEN
    FOR i = 1 TO radius_edited - 1 STEP 1
        IF radial_distance(radius_edited) = radial_distance(i) THEN
            radial_distance(radius_edited) = 0
            radius_zero = TRUE
            EXIT FOR
        END IF
    NEXT i
END IF

IF NOT radius_zero THEN
    FOR i = radius_edited + 1 TO max_number_of_radii STEP 1
        IF radial_distance(radius_edited) = radial_distance(i) THEN
            radial_distance(radius_edited) = 0
            radius_zero = TRUE
            EXIT FOR
        END IF
    NEXT i
END IF

FOR i = 1 TO max_number_of_radii - 1 STEP 1
    FOR j = i + 1 TO max_number_of_radii STEP 1
        IF (radial_distance(j) <> 0) THEN
            IF ((radial_distance(i) > radial_distance(j)) OR (radial_distance(i)
= 0)) THEN
                temp = radial_distance(i)
                radial_distance(i) = radial_distance(j)
                radial_distance(j) = temp
                IF (NOT radius_zero) THEN
                    IF i = radius_edited THEN
                        radius_edited = j
                    ELSEIF j = radius_edited THEN
                        radius_edited = i
                    END IF
                END IF
            END IF
        END IF
    NEXT j
NEXT i

i = 1
zero_found = FALSE

WHILE ((NOT zero_found) AND (i <= max_number_of_radii))
    IF radial_distance(i) = 0 THEN
        zero_found = TRUE
    ELSE

```

```

        i = i + 1
    END IF
WEND

number_of_radii = i - 1

IF radius_edited < 3 THEN
    radius_edited = 3
END IF

IF number_of_radii <= max_number_of_radii THEN
    lblRadius1.caption = RIGHT$(STR$(radius_edited - 2), 2)
    txtRadial_distance1.text = RIGHT$("    " +
FORMAT$(radial_distance(radius_edited - 2), "0.0000;0.0000;      "), 9)
    lblRadius2.caption = RIGHT$(STR$(radius_edited - 1), 2)
    txtRadial_distance2.text = RIGHT$("    " +
FORMAT$(radial_distance(radius_edited - 1), "0.0000;0.0000;      "), 9)
    lblRadius3.caption = RIGHT$(STR$(radius_edited), 2)
    txtRadial_distance3.text = RIGHT$("    " +
FORMAT$(radial_distance(radius_edited), "0.0000;0.0000;      "), 9)

    vsbRadius_number.value = radius_edited - 2
ELSE
    lblRadius1.caption = RIGHT$(STR$(radius_edited - 3), 2)
    txtRadial_distance1.text = RIGHT$("    " +
FORMAT$(radial_distance(radius_edited - 3), "0.0000;0.0000;      "), 9)
    lblRadius2.caption = RIGHT$(STR$(radius_edited - 2), 2)
    txtRadial_distance2.text = RIGHT$("    " +
FORMAT$(radial_distance(radius_edited - 2), "0.0000;0.0000;      "), 9)
    lblRadius3.caption = RIGHT$(STR$(radius_edited - 1), 2)
    txtRadial_distance3.text = RIGHT$("    " +
FORMAT$(radial_distance(radius_edited - 1), "0.0000;0.0000;      "), 9)

    vsbRadius_number.value = radius_edited - 3
END IF

IF (VAL(frmProblem_data.lblNumber_of_regions.tag) <> number_of_radii) THEN
    CALL default_regions
    CALL paint_region(0, 0)
END IF

END SUB

SUB txtSite_File_Name_KeyPress (KeyAscii AS INTEGER)

    'This routine marks the problem form as modified, and if the
    'enter key is pressed, it attempts to open the site file and read in
    'the appropriate site data.

    DIM site_file_name AS STRING

    frmMain.mnuSave_ProblemCom.Enabled = TRUE
    frmMain.mnuSave_As_ProblemCom.Enabled = TRUE

    IF (INSTR(frmProblem_data.caption, " (Modified)") = 0) THEN
        frmProblem_data.caption = frmProblem_data.caption + " (Modified)"
    END IF

    IF KeyAscii = KEY_RETURN THEN
        KeyAscii = 0
        site_file_name = txtSite_file_Name.text
        'open_site_from_problem (site_file_name)
    END IF

```

```

END SUB

SUB txtSite_File_Name_LostFocus ()
    'Check and see if the site file name is correct, and if it is read
    'in the site data.

    DIM site_file_name AS STRING

    IF frmProblem_data.VISIBLE THEN
        IF (screen.activecontrol.tag <> "Browse") AND (screen.activecontrol.tag <>
    "Hiding Form") AND (screen.activecontrol.tag <> "Open") AND
    (screen.activecontrol.tag <> "Close") THEN
            site_file_name = txtSite_file_Name.text
            'open_site_from_problem (site_file_name)
        END IF
        IF screen.activecontrol.tag = "Hiding Form" THEN
            screen.activecontrol.tag = ""
        END IF

    END IF

    END SUB

SUB vsbRadius_Number_Change ()
    'This function updates the problem form when a user has changed it.
    'It simply puts the correct values for radii back in the correct places.
    'It is called whenever the scroll bar on the radii box is moved.

    DIM radius_number(3) AS STRING
    DIM i AS INTEGER

    IF (vsbRadius_number.value <= max_number_of_radii - 2) THEN
        radius_number(1) = RIGHT$(STR$(vsbRadius_number.value), 2)
        radius_number(2) = RIGHT$(STR$(vsbRadius_number.value + 1), 2)
        radius_number(3) = RIGHT$(STR$(vsbRadius_number.value + 2), 2)

        lblRadius1.caption = radius_number(1)
        txtRadial_distance1.text = RIGHT$(" " +
FORMAT$(radial_distance(vsbRadius_number.value), "0.0000;0.0000;      "), 9)

        lblRadius2.caption = radius_number(2)
        txtRadial_distance2.text = RIGHT$(" " +
FORMAT$(radial_distance(vsbRadius_number.value + 1), "0.0000;0.0000;      "), 9)

        lblRadius3.caption = radius_number(3)
        txtRadial_distance3.text = RIGHT$(" " +
FORMAT$(radial_distance(vsbRadius_number.value + 2), "0.0000;0.0000;      "), 9)
    END IF

    END SUB

```

**FILENAME: REGION.FR**

```
Version 1.00
BEGIN Form frmRegion
    AutoRedraw      = -1
    BackColor       = QBColor(3)
    BorderStyle     = 0
    Caption         = "Region"
    ControlBox      = -1
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(22)
    Left            = Char(0)
    MaxButton       = -1
    MinButton       = -1
    MousePointer    = 0
    Tag             = ""
    Top             = Char(0)
    Visible         = -1
    Width           = Char(78)
    WindowState     = 0
    BEGIN Frame Frame1
        BackColor      = QBColor(3)
        Caption         = "Regions"
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(3)
        Left            = Char(64)
        MousePointer    = 0
        TabIndex        = 74
        Tag             = ""
        Top             = Char(1)
        Visible         = -1
        Width           = Char(11)
        BEGIN Label lblnum_regions
            Alignment      = 1
            AutoSize        = 0
            BackColor       = QBColor(3)
            BorderStyle     = 0
            Caption         = "99"
            DragMode        = 0
            Enabled         = -1
            ForeColor       = QBColor(0)
            Height          = Char(1)
            Left            = Char(5)
            MousePointer    = 0
            TabIndex        = 75
            Tag             = ""
            Top             = Char(0)
            Visible         = -1
            Width           = Char(3)
        END
    END
    BEGIN CommandButton cmdSort
        BackColor       = QBColor(3)
        Cancel          = 0
        Caption         = "&Sort"
        Default         = 0
        DragMode        = 0
        Enabled         = -1
        Height          = Char(3)
    END
```

```

        Left          = Char(64)
        MousePointer = 0
        TabIndex     = 4
        TabStop      = -1
        Tag          = ""
        Top          = Char(10)
        Visible      = -1
        Width        = Char(11)

    END
    BEGIN CommandButton cmdDefault
        BackColor    = QBColor(3)
        Cancel       = 0
        Caption      = "Deefault"
        Default      = 0
        DragMode     = 0
        Enabled      = -1
        Height       = Char(3)
        Left         = Char(64)
        MousePointer = 0
        TabIndex     = 5
        TabStop      = -1
        Tag          = ""
        Top          = Char(13)
        Visible      = -1
        Width        = Char(11)

    END
    BEGIN CommandButton cmdClear
        BackColor    = QBColor(3)
        Cancel       = 0
        Caption      = "Clea&r"
        Default      = 0
        DragMode     = 0
        Enabled      = -1
        Height       = Char(3)
        Left         = Char(64)
        MousePointer = 0
        TabIndex     = 6
        TabStop      = -1
        Tag          = ""
        Top          = Char(16)
        Visible      = -1
        Width        = Char(11)

    END
    BEGIN CommandButton cmdClose
        BackColor    = QBColor(3)
        Cancel       = 0
        Caption      = "&Close"
        Default      = 0
        DragMode     = 0
        Enabled      = -1
        Height       = Char(3)
        Left         = Char(64)
        MousePointer = 0
        TabIndex     = 7
        TabStop      = -1
        Tag          = ""
        Top          = Char(19)
        Visible      = -1
        Width        = Char(11)

    END
    BEGIN Frame Frame2
        BackColor    = QBColor(3)
        Caption      = "Radius"

```

```

DragMode      = 0
Enabled       = -1
ForeColor     = QBColor(0)
Height        = Char(3)
Left          = Char(64)
MousePointer  = 0
TabIndex      = 76
Tag           = ""
Top           = Char(5)
Visible       = -1
Width         = Char(11)
END
BEGIN Label lblSectors
    Alignment    = 0
    AutoSize     = -1
    BackColor    = QBColor(3)
    BorderStyle  = 0
    Caption      = "Sectors"
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(1)
    Left         = Char(4)
    MousePointer = 0
   TabIndex      = 71
    Tag          = ""
    Top          = Char(3)
    Visible      = -1
    Width        = Char(7)
END
BEGIN CommandButton cmdUp
    BackColor    = QBColor(3)
    Cancel       = 0
    Caption      = "&Up"
    Default      = 0
    DragMode     = 0
    Enabled      = -1
    Height       = Char(3)
    Left         = Char(33)
    MousePointer = 0
   TabIndex      = 2
    TabStop      = -1
    Tag          = ""
    Top          = Char(19)
    Visible      = -1
    Width        = Char(11)
END
BEGIN CommandButton cmdRight
    BackColor    = QBColor(3)
    Cancel       = 0
    Caption      = "&Right"
    Default      = 0
    DragMode     = 0
    Enabled      = -1
    Height       = Char(3)
    Left         = Char(21)
    MousePointer = 0
   TabIndex      = 1
    TabStop      = -1
    Tag          = ""
    Top          = Char(19)
    Visible      = -1
    Width        = Char(11)

```

```

END
BEGIN CommandButton cmdDown
    BackColor      = QBColor(3)
    Cancel        = 0
    Caption        = "&Down"
    Default        = 0
    DragMode       = 0
    Enabled        = -1
    Height         = Char(3)
    Left           = Char(45)
    MousePointer  = 0
    TabIndex       = 3
    TabStop        = -1
    Tag            = ""
    Top            = Char(19)
    Visible        = -1
    Width          = Char(11)
END
BEGIN CommandButton cmdLeft
    BackColor      = QBColor(3)
    Cancel        = 0
    Caption        = "&Left"
    Default        = 0
    DragMode       = 0
    Enabled        = -1
    Height         = Char(3)
    Left           = Char(9)
    MousePointer  = 0
    TabIndex       = 0
    TabStop        = -1
    Tag            = ""
    Top            = Char(19)
    Visible        = -1
    Width          = Char(11)
END
BEGIN Label lblRadial_Distances
    Alignment      = 0
    AutoSize       = -1
    BackColor      = QBColor(3)
    BorderStyle    = 0
    Caption        = "Radial Distances in "
    DragMode       = 0
    Enabled        = -1
    ForeColor      = QBColor(0)
    Height         = Char(1)
    Left           = Char(24)
    MousePointer  = 0
    TabIndex       = 72
    Tag            = ""
    Top            = Char(1)
    Visible        = -1
    Width          = Char(20)
END
BEGIN Label lblUnits
    Alignment      = 0
    AutoSize       = -1
    BackColor      = QBColor(3)
    BorderStyle    = 0
    Caption        = "Miles"
    DragMode       = 0
    Enabled        = -1
    ForeColor      = QBColor(0)
    Height         = Char(1)

```

```

        Left      = Char(44)
        MousePointer = 0
        TabIndex    = 73
        Tag         = ""
        Top         = Char(1)
        Visible     = -1
        Width       = Char(5)
END
BEGIN Label lblSector
        Alignment  = 2
        AutoSize   = 0
        BackColor  = QBColor(3)
        BorderStyle = 1
        Caption    = " "
        DragMode   = 0
        Enabled    = -1
        ForeColor  = QBColor(0)
        Height     = Char(3)
        Index      = 0
        Left       = Char(3)
        MousePointer = 0
        TabIndex   = 8
        Tag        = ""
        Top        = Char(4)
        Visible    = -1
        Width      = Char(10)
END
BEGIN Label lblRdist
        Alignment  = 1
        AutoSize   = 0
        BackColor  = QBColor(3)
        BorderStyle = 1
        Caption    = " "
        DragMode   = 0
        Enabled    = -1
        ForeColor  = QBColor(0)
        Height     = Char(3)
        Index      = 0
        Left       = Char(12)
        MousePointer = 0
        TabIndex   = 15
        Tag        = ""
        Top        = Char(2)
        Visible    = -1
        Width      = Char(8)
END
BEGIN Label lblRdist
        Alignment  = 1
        AutoSize   = 0
        BackColor  = QBColor(3)
        BorderStyle = 1
        Caption    = " "
        DragMode   = 0
        Enabled    = -1
        ForeColor  = QBColor(0)
        Height     = Char(3)
        Index      = 1
        Left       = Char(19)
        MousePointer = 0
        TabIndex   = 16
        Tag        = ""
        Top        = Char(2)
        Visible    = -1

```

```

        Width      = Char(8)
END
BEGIN Label lblRdist
    Alignment  = 1
    AutoSize   = 0
    BackColor  = QBColor(3)
    BorderStyle= 1
    Caption    = " "
    DragMode   = 0
    Enabled    = -1
    ForeColor  = QBColor(0)
    Height     = Char(3)
    Index      = 2
    Left       = Char(26)
    MousePointer = 0
    TabIndex   = 17
    Tag        = ""
    Top        = Char(2)
    Visible   = -1
    Width      = Char(8)
END
BEGIN Label lblRdist
    Alignment  = 1
    AutoSize   = 0
    BackColor  = QBColor(3)
    BorderStyle= 1
    Caption    = " "
    DragMode   = 0
    Enabled    = -1
    ForeColor  = QBColor(0)
    Height     = Char(3)
    Index      = 3
    Left       = Char(33)
    MousePointer = 0
    TabIndex   = 18
    Tag        = ""
    Top        = Char(2)
    Visible   = -1
    Width      = Char(8)
END
BEGIN Label lblRdist
    Alignment  = 1
    AutoSize   = 0
    BackColor  = QBColor(3)
    BorderStyle= 1
    Caption    = " "
    DragMode   = 0
    Enabled    = -1
    ForeColor  = QBColor(0)
    Height     = Char(3)
    Index      = 4
    Left       = Char(40)
    MousePointer = 0
    TabIndex   = 19
    Tag        = ""
    Top        = Char(2)
    Visible   = -1
    Width      = Char(8)
END
BEGIN Label lblRdist
    Alignment  = 1
    AutoSize   = 0
    BackColor  = QBColor(3)

```

```

BorderStyle = 1
Caption = " "
DragMode = 0
Enabled = -1
ForeColor = QBColor(0)
Height = Char(3)
Index = 5
Left = Char(47)
MousePointer = 0
TabIndex = 20
Tag = ""
Top = Char(2)
Visible = -1
Width = Char(8)

END
BEGIN Label lblRdist
    Alignment = 1
    AutoSize = 0
    BackColor = QBColor(3)
    BorderStyle = 1
    Caption = " "
    DragMode = 0
    Enabled = -1
    ForeColor = QBColor(0)
    Height = Char(3)
    Index = 6
    Left = Char(54)
    MousePointer = 0
    TabIndex = 21
    Tag = ""
    Top = Char(2)
    Visible = -1
    Width = Char(8)

END
BEGIN TextBox txtsector
    BackColor = QBColor(3)
    BorderStyle = 1
    DragMode = 0
    Enabled = -1
    ForeColor = QBColor(0)
    Height = Char(3)
    Index = 0
    Left = Char(12)
    MousePointer = 0
    MultiLine = 0
    ScrollBars = 0
    TabIndex = 22
    TabStop = -1
    Tag = ""
    Text = ""
    Top = Char(4)
    Visible = -1
    Width = Char(8)

END
BEGIN TextBox txtsector
    BackColor = QBColor(3)
    BorderStyle = 1
    DragMode = 0
    Enabled = -1
    ForeColor = QBColor(0)
    Height = Char(3)
    Index = 1
    Left = Char(19)

```

```

MousePointer = 0
MultiLine    = 0
ScrollBars   = 0
TabIndex     = 23
TabStop      = -1
Tag          = ""
Text         = " "
Top          = Char(4)
Visible      = -1
Width        = Char(8)

END
BEGIN TextBox txtsector
  BackColor   = QBColor(3)
  BorderStyle = 1
  DragMode    = 0
  Enabled     = -1
  ForeColor   = QBColor(0)
  Height      = Char(3)
  Index       = 2
  Left         = Char(26)
  MousePointer = 0
  MultiLine    = 0
  ScrollBars   = 0
  TabIndex     = 24
  TabStop      = -1
  Tag          = ""
  Text         = " "
  Top          = Char(4)
  Visible      = -1
  Width        = Char(8)

END
BEGIN TextBox txtsector
  BackColor   = QBColor(3)
  BorderStyle = 1
  DragMode    = 0
  Enabled     = -1
  ForeColor   = QBColor(0)
  Height      = Char(3)
  Index       = 3
  Left         = Char(33)
  MousePointer = 0
  MultiLine    = 0
  ScrollBars   = 0
  TabIndex     = 25
  TabStop      = -1
  Tag          = ""
  Text         = " "
  Top          = Char(4)
  Visible      = -1
  Width        = Char(8)

END
BEGIN TextBox txtsector
  BackColor   = QBColor(3)
  BorderStyle = 1
  DragMode    = 0
  Enabled     = -1
  ForeColor   = QBColor(0)
  Height      = Char(3)
  Index       = 4
  Left         = Char(40)
  MousePointer = 0
  MultiLine    = 0
  ScrollBars   = 0

```

```

TabIndex      = 26
TabStop       = -1
Tag           = ""
Text          = " "
Top           = Char(4)
Visible       = -1
Width         = Char(8)
END
BEGIN TextBox txtsector
BackColor     = QBColor(3)
BorderStyle   = 1
DragMode      = 0
Enabled       = -1
ForeColor     = QBColor(0)
Height        = Char(3)
Index         = 5
Left          = Char(47)
MousePointer  = 0
MultiLine     = 0
ScrollBars    = 0
TabIndex      = 27
TabStop       = -1
Tag           = ""
Text          = " "
Top           = Char(4)
Visible       = -1
Width         = Char(8)
END
BEGIN TextBox txtsector
BackColor     = QBColor(3)
BorderStyle   = 1
DragMode      = 0
Enabled       = -1
ForeColor     = QBColor(0)
Height        = Char(3)
Index         = 6
Left          = Char(54)
MousePointer  = 0
MultiLine     = 0
ScrollBars    = 0
TabIndex      = 28
TabStop       = -1
Tag           = ""
Text          = " "
Top           = Char(4)
Visible       = -1
Width         = Char(8)
END
BEGIN Label lblSector
Alignment     = 2
AutoSize      = 0
BackColor     = QBColor(3)
BorderStyle   = 1
Caption       = " "
DragMode      = 0
Enabled       = -1
ForeColor     = QBColor(0)
Height        = Char(3)
Index         = 1
Left          = Char(3)
MousePointer  = 0
TabIndex      = 9
Tag           = ""

```

```

        Top          = Char(6)
        Visible      = -1
        Width        = Char(10)
    END
    BEGIN Label lblSector
        Alignment    = 2
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 1
        Caption      = " "
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 2
        Left         = Char(3)
        MousePointer = 0
        TabIndex     = 10
        Tag          = ""
        Top          = Char(8)
        Visible      = -1
        Width        = Char(10)
    END
    BEGIN Label lblSector
        Alignment    = 2
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 1
        Caption      = " "
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 3
        Left         = Char(3)
        MousePointer = 0
        TabIndex     = 11
        Tag          = ""
        Top          = Char(10)
        Visible      = -1
        Width        = Char(10)
    END
    BEGIN Label lblSector
        Alignment    = 2
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 1
        Caption      = " "
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 4
        Left         = Char(3)
        MousePointer = 0
        TabIndex     = 12
        Tag          = ""
        Top          = Char(12)
        Visible      = -1
        Width        = Char(10)
    END
    BEGIN Label lblSector
        Alignment    = 2

```

```

AutoSize      = 0
BackColor    = QBColor(3)
BorderStyle  = 1
Caption      = " "
DragMode     = 0
Enabled      = -1
ForeColor    = QBColor(0)
Height       = Char(3)
Index        = 5
Left         = Char(3)
MousePointer = 0
TabIndex     = 13
Tag          = ""
Top          = Char(14)
Visible      = -1
Width        = Char(10)
END
BEGIN Label lblSector
Alignment    = 2
AutoSize     = 0
BackColor   = QBColor(3)
BorderStyle  = 1
Caption     = " "
DragMode    = 0
Enabled     = -1
ForeColor   = QBColor(0)
Height      = Char(3)
Index       = 6
Left        = Char(3)
MousePointer = 0
TabIndex    = 14
Tag          = ""
Top         = Char(16)
Visible     = -1
Width        = Char(10)
END
BEGIN TextBox txtsector
BackColor   = QBColor(3)
BorderStyle = 1
DragMode    = 0
Enabled     = -1
ForeColor   = QBColor(0)
Height      = Char(3)
Index       = 7
Left        = Char(12)
MousePointer = 0
MultiLine    = 0
ScrollBars  = 0
TabIndex    = 29
TabStop     = -1
Tag          = ""
Text         = " "
Top          = Char(6)
Visible     = -1
Width        = Char(8)
END
BEGIN TextBox txtsector
BackColor   = QBColor(3)
BorderStyle = 1
DragMode    = 0
Enabled     = -1
ForeColor   = QBColor(0)
Height      = Char(3)

```

```

Index      = 8
Left       = Char(19)
MousePointer = 0
MultiLine   = 0
ScrollBars  = 0
TabIndex    = 30
TabStop     = -1
Tag         = ""
Text        = " "
Top         = Char(6)
Visible     = -1
Width       = Char(8)

END
BEGIN TextBox txtsector
BackColor   = QBColor(3)
BorderStyle = 1
DragMode    = 0
Enabled     = -1
ForeColor   = QBColor(0)
Height      = Char(3)
Index       = 9
Left        = Char(26)
MousePointer = 0
MultiLine   = 0
ScrollBars  = 0
TabIndex    = 31
TabStop     = -1
Tag         = ""
Text        = " "
Top         = Char(6)
Visible     = -1
Width       = Char(8)

END
BEGIN TextBox txtsector
BackColor   = QBColor(3)
BorderStyle = 1
DragMode    = 0
Enabled     = -1
ForeColor   = QBColor(0)
Height      = Char(3)
Index       = 10
Left        = Char(33)
MousePointer = 0
MultiLine   = 0
ScrollBars  = 0
TabIndex    = 32
TabStop     = -1
Tag         = ""
Text        = " "
Top         = Char(6)
Visible     = -1
Width       = Char(8)

END
BEGIN TextBox txtsector
BackColor   = QBColor(3)
BorderStyle = 1
DragMode    = 0
Enabled     = -1
ForeColor   = QBColor(0)
Height      = Char(3)
Index       = 11
Left        = Char(40)
MousePointer = 0

```

```

MultiLine      = 0
ScrollBars    = 0
TabIndex      = 33
TabStop       = -1
Tag           = ""
Text          = " "
Top           = Char(6)
Visible       = -1
Width         = Char(8)

END
BEGIN TextBox txtsector
BackColor      = QBColor(3)
BorderStyle    = 1
DragMode       = 0
Enabled        = -1
ForeColor      = QBColor(0)
Height         = Char(3)
Index          = 12
Left           = Char(47)
MousePointer   = 0
MultiLine      = 0
ScrollBars    = 0
TabIndex      = 34
TabStop       = -1
Tag           = ""
Text          = " "
Top           = Char(6)
Visible       = -1
Width         = Char(8)

END
BEGIN TextBox txtsector
BackColor      = QBColor(3)
BorderStyle    = 1
DragMode       = 0
Enabled        = -1
ForeColor      = QBColor(0)
Height         = Char(3)
Index          = 13
Left           = Char(54)
MousePointer   = 0
MultiLine      = 0
ScrollBars    = 0
TabIndex      = 35
TabStop       = -1
Tag           = ""
Text          = " "
Top           = Char(6)
Visible       = -1
Width         = Char(8)

END
BEGIN TextBox txtsector
BackColor      = QBColor(3)
BorderStyle    = 1
DragMode       = 0
Enabled        = -1
ForeColor      = QBColor(0)
Height         = Char(3)
Index          = 14
Left           = Char(12)
MousePointer   = 0
MultiLine      = 0
ScrollBars    = 0
TabIndex      = 36

```

```

    TabStop      = -1
    Tag          = ""
    Text         = " "
    Top          = Char(8)
    Visible     = -1
    Width        = Char(8)
END
BEGIN TextBox txtsector
    BackColor    = QBColor(3)
    BorderStyle  = 1
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(3)
    Index        = 15
    Left          = Char(19)
    MousePointer = 0
    MultiLine    = 0
    ScrollBars   = 0
    TabIndex     = 37
    TabStop      = -1
    Tag          = ""
    Text         = " "
    Top          = Char(8)
    Visible     = -1
    Width        = Char(8)
END
BEGIN TextBox txtsector
    BackColor    = QBColor(3)
    BorderStyle  = 1
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(3)
    Index        = 16
    Left          = Char(26)
    MousePointer = 0
    MultiLine    = 0
    ScrollBars   = 0
    TabIndex     = 38
    TabStop      = -1
    Tag          = ""
    Text         = " "
    Top          = Char(8)
    Visible     = -1
    Width        = Char(8)
END
BEGIN TextBox txtsector
    BackColor    = QBColor(3)
    BorderStyle  = 1
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(3)
    Index        = 17
    Left          = Char(33)
    MousePointer = 0
    MultiLine    = 0
    ScrollBars   = 0
    TabIndex     = 39
    TabStop      = -1
    Tag          = ""
    Text         = " "

```

```

        Top          = Char(8)
        Visible     = -1
        Width       = Char(8)
END
BEGIN TextBox txtsector
        BackColor    = QBColor(3)
        BorderStyle  = 1
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 18
        Left         = Char(40)
        MousePointer = 0
        MultiLine    = 0
        ScrollBars   = 0
        TabIndex     = 40
        TabStop      = -1
        Tag          = ""
        Text         = " "
        Top          = Char(8)
        Visible     = -1
        Width       = Char(8)
END
BEGIN TextBox txtsector
        BackColor    = QBColor(3)
        BorderStyle  = 1
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 19
        Left         = Char(47)
        MousePointer = 0
        MultiLine    = 0
        ScrollBars   = 0
        TabIndex     = 41
        TabStop      = -1
        Tag          = ""
        Text         = " "
        Top          = Char(8)
        Visible     = -1
        Width       = Char(8)
END
BEGIN TextBox txtsector
        BackColor    = QBColor(3)
        BorderStyle  = 1
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 20
        Left         = Char(54)
        MousePointer = 0
        MultiLine    = 0
        ScrollBars   = 0
        TabIndex     = 42
        TabStop      = -1
        Tag          = ""
        Text         = " "
        Top          = Char(8)
        Visible     = -1
        Width       = Char(8)

```

```

END
BEGIN TextBox txtsector
    BackColor      = QBColor(3)
    BorderStyle    = 1
    DragMode       = 0
    Enabled        = -1
    ForeColor      = QBColor(0)
    Height         = Char(3)
    Index          = 21
    Left           = Char(12)
    MousePointer   = 0
    MultiLine      = 0
    ScrollBars     = 0
    TabIndex       = 43
    TabStop        = -1
    Tag            = ""
    Text           = " "
    Top            = Char(10)
    Visible        = -1
    Width          = Char(8)
END
BEGIN TextBox txtsector
    BackColor      = QBColor(3)
    BorderStyle    = 1
    DragMode       = 0
    Enabled        = -1
    ForeColor      = QBColor(0)
    Height         = Char(3)
    Index          = 22
    Left           = Char(19)
    MousePointer   = 0
    MultiLine      = 0
    ScrollBars     = 0
    TabIndex       = 44
    TabStop        = -1
    Tag            = ""
    Text           = " "
    Top            = Char(10)
    Visible        = -1
    Width          = Char(8)
END
BEGIN TextBox txtsector
    BackColor      = QBColor(3)
    BorderStyle    = 1
    DragMode       = 0
    Enabled        = -1
    ForeColor      = QBColor(0)
    Height         = Char(3)
    Index          = 23
    Left           = Char(26)
    MousePointer   = 0
    MultiLine      = 0
    ScrollBars     = 0
    TabIndex       = 45
    TabStop        = -1
    Tag            = ""
    Text           = " "
    Top            = Char(10)
    Visible        = -1
    Width          = Char(8)
END
BEGIN TextBox txtsector
    BackColor      = QBColor(3)

```

```

BorderStyle = 1
DragMode = 0
Enabled = -1
ForeColor = QBColor(0)
Height = Char(3)
Index = 24
Left = Char(33)
MousePointer = 0
MultiLine = 0
ScrollBars = 0
TabIndex = 46
TabStop = -1
Tag = ""
Text = " "
Top = Char(10)
Visible = -1
Width = Char(8)

END
BEGIN TextBox txtsector
BackColor = QBColor(3)
BorderStyle = 1
DragMode = 0
Enabled = -1
ForeColor = QBColor(0)
Height = Char(3)
Index = 25
Left = Char(40)
MousePointer = 0
MultiLine = 0
ScrollBars = 0
TabIndex = 47
TabStop = -1
Tag = ""
Text = " "
Top = Char(10)
Visible = -1
Width = Char(8)

END
BEGIN TextBox txtsector
BackColor = QBColor(3)
BorderStyle = 1
DragMode = 0
Enabled = -1
ForeColor = QBColor(0)
Height = Char(3)
Index = 26
Left = Char(47)
MousePointer = 0
MultiLine = 0
ScrollBars = 0
TabIndex = 48
TabStop = -1
Tag = ""
Text = " "
Top = Char(10)
Visible = -1
Width = Char(8)

END
BEGIN TextBox txtsector
BackColor = QBColor(3)
BorderStyle = 1
DragMode = 0
Enabled = -1

```

```

ForeColor      = QBColor(0)
Height        = Char(3)
Index         = 27
Left          = Char(54)
MousePointer  = 0
MultiLine     = 0
ScrollBars    = 0
TabIndex      = 49
TabStop       = -1
Tag           = ""
Text          = " "
Top           = Char(10)
Visible       = -1
Width         = Char(8)

END
BEGIN TextBox txtsector
  BackColor    = QBColor(3)
  BorderStyle  = 1
  DragMode     = 0
  Enabled      = -1
  ForeColor    = QBColor(0)
  Height       = Char(3)
  Index        = 28
  Left          = Char(12)
  MousePointer = 0
  MultiLine    = 0
  ScrollBars   = 0
 TabIndex      = 50
  TabStop      = -1
  Tag          = ""
  Text          = " "
  Top           = Char(12)
  Visible       = -1
  Width         = Char(8)

END
BEGIN TextBox txtsector
  BackColor    = QBColor(3)
  BorderStyle  = 1
  DragMode     = 0
  Enabled      = -1
  ForeColor    = QBColor(0)
  Height       = Char(3)
  Index        = 29
  Left          = Char(19)
  MousePointer = 0
  MultiLine    = 0
  ScrollBars   = 0
 TabIndex      = 51
  TabStop      = -1
  Tag          = ""
  Text          = " "
  Top           = Char(12)
  Visible       = -1
  Width         = Char(8)

END
BEGIN TextBox txtsector
  BackColor    = QBColor(3)
  BorderStyle  = 1
  DragMode     = 0
  Enabled      = -1
  ForeColor    = QBColor(0)
  Height       = Char(3)
  Index        = 30

```

```

        Left      = Char(26)
        MousePointer = 0
        MultiLine   = 0
        ScrollBars  = 0
        TabIndex    = 52
        TabStop     = -1
        Tag         = ""
        Text        = " "
        Top         = Char(12)
        Visible    = -1
        Width       = Char(8)
    END
    BEGIN TextBox txtsector
        BackColor  = QBColor(3)
        BorderStyle = 1
        DragMode   = 0
        Enabled    = -1
        ForeColor  = QBColor(0)
        Height     = Char(3)
        Index      = 31
        Left        = Char(33)
        MousePointer = 0
        MultiLine   = 0
        ScrollBars  = 0
        TabIndex    = 53
        TabStop     = -1
        Tag         = ""
        Text        = " "
        Top         = Char(12)
        Visible    = -1
        Width       = Char(8)
    END
    BEGIN TextBox txtsector
        BackColor  = QBColor(3)
        BorderStyle = 1
        DragMode   = 0
        Enabled    = -1
        ForeColor  = QBColor(0)
        Height     = Char(3)
        Index      = 32
        Left        = Char(40)
        MousePointer = 0
        MultiLine   = 0
        ScrollBars  = 0
        TabIndex    = 54
        TabStop     = -1
        Tag         = ""
        Text        = " "
        Top         = Char(12)
        Visible    = -1
        Width       = Char(8)
    END
    BEGIN TextBox txtsector
        BackColor  = QBColor(3)
        BorderStyle = 1
        DragMode   = 0
        Enabled    = -1
        ForeColor  = QBColor(0)
        Height     = Char(3)
        Index      = 33
        Left        = Char(47)
        MousePointer = 0
        MultiLine   = 0
    END

```

```

ScrollBars    = 0
TabIndex      = 55
TabStop       = -1
Tag           = ""
Text          = " "
Top           = Char(12)
Visible       = -1
Width         = Char(8)

END
BEGIN TextBox txtsector
  BackColor    = QBColor(3)
  BorderStyle  = 1
  DragMode     = 0
  Enabled      = -1
  ForeColor    = QBColor(0)
  Height       = Char(3)
  Index        = 34
  Left          = Char(54)
  MousePointer = 0
  MultiLine    = 0
  ScrollBars   = 0
  TabIndex     = 56
  TabStop      = -1
  Tag          = ""
  Text          = " "
  Top           = Char(12)
  Visible       = -1
  Width         = Char(8)
END
BEGIN TextBox txtsector
  BackColor    = QBColor(3)
  BorderStyle  = 1
  DragMode     = 0
  Enabled      = -1
  ForeColor    = QBColor(0)
  Height       = Char(3)
  Index        = 35
  Left          = Char(12)
  MousePointer = 0
  MultiLine    = 0
  ScrollBars   = 0
  TabIndex     = 57
  TabStop      = -1
  Tag          = ""
  Text          = " "
  Top           = Char(14)
  Visible       = -1
  Width         = Char(8)
END
BEGIN TextBox txtsector
  BackColor    = QBColor(3)
  BorderStyle  = 1
  DragMode     = 0
  Enabled      = -1
  ForeColor    = QBColor(0)
  Height       = Char(3)
  Index        = 36
  Left          = Char(19)
  MousePointer = 0
  MultiLine    = 0
  ScrollBars   = 0
  TabIndex     = 58
  TabStop      = -1

```

```

        Tag      = ""
        Text    = " "
        Top     = Char(14)
        Visible = -1
        Width   = Char(8)
    END
    BEGIN TextBox txtsector
        BackColor = QBColor(3)
        BorderStyle = 1
        DragMode = 0
        Enabled = -1
        ForeColor = QBColor(0)
        Height = Char(3)
        Index = 37
        Left = Char(26)
        MousePointer = 0
        MultiLine = 0
        ScrollBars = 0
        TabIndex = 59
        TabStop = -1
        Tag = ""
        Text = " "
        Top = Char(14)
        Visible = -1
        Width = Char(8)
    END
    BEGIN TextBox txtsector
        BackColor = QBColor(3)
        BorderStyle = 1
        DragMode = 0
        Enabled = -1
        ForeColor = QBColor(0)
        Height = Char(3)
        Index = 38
        Left = Char(33)
        MousePointer = 0
        MultiLine = 0
        ScrollBars = 0
        TabIndex = 60
        TabStop = -1
        Tag = ""
        Text = " "
        Top = Char(14)
        Visible = -1
        Width = Char(8)
    END
    BEGIN TextBox txtsector
        BackColor = QBColor(3)
        BorderStyle = 1
        DragMode = 0
        Enabled = -1
        ForeColor = QBColor(0)
        Height = Char(3)
        Index = 39
        Left = Char(40)
        MousePointer = 0
        MultiLine = 0
        ScrollBars = 0
        TabIndex = 61
        TabStop = -1
        Tag = ""
        Text = " "
        Top = Char(14)
    END

```

```

        Visible      = -1
        Width       = Char(8)
END
BEGIN TextBox txtsector
        BackColor    = QBColor(3)
        BorderStyle  = 1
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 40
        Left         = Char(47)
        MousePointer = 0
        MultiLine    = 0
        ScrollBars   = 0
        TabIndex     = 62
        TabStop      = -1
        Tag          = ""
        Text         = " "
        Top          = Char(14)
        Visible      = -1
        Width        = Char(8)
END
BEGIN TextBox txtsector
        BackColor    = QBColor(3)
        BorderStyle  = 1
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 41
        Left         = Char(54)
        MousePointer = 0
        MultiLine    = 0
        ScrollBars   = 0
        TabIndex     = 63
        TabStop      = -1
        Tag          = ""
        Text         = " "
        Top          = Char(14)
        Visible      = -1
        Width        = Char(8)
END
BEGIN TextBox txtsector
        BackColor    = QBColor(3)
        BorderStyle  = 1
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 42
        Left         = Char(12)
        MousePointer = 0
        MultiLine    = 0
        ScrollBars   = 0
        TabIndex     = 64
        TabStop      = -1
        Tag          = ""
        Text         = " "
        Top          = Char(16)
        Visible      = -1
        Width        = Char(8)
END

```

```

BEGIN TextBox txtsector
  BackColor      = QBColor(3)
  BorderStyle    = 1
  DragMode       = 0
  Enabled        = -1
  ForeColor      = QBColor(0)
  Height         = Char(3)
  Index          = 43
  Left           = Char(19)
  MousePointer   = 0
  MultiLine      = 0
  ScrollBars     = 0
  TabIndex       = 65
  TabStop        = -1
  Tag            = ""
  Text           = " "
  Top            = Char(16)
  Visible        = -1
  Width          = Char(8)
END
BEGIN TextBox txtsector
  BackColor      = QBColor(3)
  BorderStyle    = 1
  DragMode       = 0
  Enabled        = -1
  ForeColor      = QBColor(0)
  Height         = Char(3)
  Index          = 44
  Left           = Char(26)
  MousePointer   = 0
  MultiLine      = 0
  ScrollBars     = 0
  TabIndex       = 66
  TabStop        = -1
  Tag            = ""
  Text           = " "
  Top            = Char(16)
  Visible        = -1
  Width          = Char(8)
END
BEGIN TextBox txtsector
  BackColor      = QBColor(3)
  BorderStyle    = 1
  DragMode       = 0
  Enabled        = -1
  ForeColor      = QBColor(0)
  Height         = Char(3)
  Index          = 45
  Left           = Char(33)
  MousePointer   = 0
  MultiLine      = 0
  ScrollBars     = 0
  TabIndex       = 67
  TabStop        = -1
  Tag            = ""
  Text           = " "
  Top            = Char(16)
  Visible        = -1
  Width          = Char(8)
END
BEGIN TextBox txtsector
  BackColor      = QBColor(3)
  BorderStyle    = 1

```

```

DragMode      = 0
Enabled       = -1
ForeColor     = QBColor(0)
Height        = Char(3)
Index         = 46
Left          = Char(40)
MousePointer  = 0
MultiLine     = 0
ScrollBars    = 0
TabIndex      = 68
TabStop       = -1
Tag           = ""
Text          = " "
Top           = Char(16)
Visible       = -1
Width         = Char(8)

END
BEGIN TextBox txtsector
  BackColor     = QBColor(3)
  BorderStyle   = 1
  DragMode      = 0
  Enabled       = -1
  ForeColor     = QBColor(0)
  Height        = Char(3)
  Index         = 47
  Left          = Char(47)
  MousePointer  = 0
  MultiLine     = 0
  ScrollBars    = 0
 TabIndex      = 69
  TabStop       = -1
  Tag           = ""
  Text          = " "
  Top           = Char(16)
  Visible       = -1
  Width         = Char(8)

END
BEGIN TextBox txtsector
  BackColor     = QBColor(3)
  BorderStyle   = 1
  DragMode      = 0
  Enabled       = -1
  ForeColor     = QBColor(0)
  Height        = Char(3)
  Index         = 48
  Left          = Char(54)
  MousePointer  = 0
  MultiLine     = 0
  ScrollBars    = 0
 TabIndex      = 70
  TabStop       = -1
  Tag           = ""
  Text          = " "
  Top           = Char(16)
  Visible       = -1
  Width         = Char(8)

END
BEGIN Label lblnum_radii
  Alignment     = 1
  AutoSize      = 0
  BackColor     = QBColor(3)
  BorderStyle   = 0
  Caption       = "50"

```

```

        DragMode      = 0
        Enabled       = -1
        ForeColor     = QBColor(0)
        Height        = Char(1)
        Left          = Char(70)
        MousePointer  = 0
        TabIndex      = 77
        Tag           = ""
        Top           = Char(6)
        Visible       = -1
        Width         = Char(3)
    END
END
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

SUB cmdClear_Click ()
    DIM i AS INTEGER, j AS INTEGER

    number_econ_regions = 0
    frmProblem_data.lblNumber_of_regions.caption = ""

    FOR i = 0 TO number_of_segments - 1
        FOR j = 2 TO max_number_of_radii
            region_index(j, i) = 0
        NEXT j
    NEXT i

    x_position = 0
    y_position = 0

    CALL paint_region(0, 0)

END SUB

SUB cmdClose_Click ()
    CALL cmdSort_Click
    IF frmRegion.cmdSort.Tag = "F" THEN
        EXIT SUB
    ELSE
        frmRegion.HIDE
    END IF

END SUB

SUB cmdDefault_Click ()
    CALL default_regions
    x_position = 0
    y_position = 0

    CALL paint_region(0, 0)

END SUB

SUB cmdDown_Click ()
    CALL paint_region(0, 1)

END SUB

```

```

SUB cmdLeft_Click ()
    CALL paint_region(-1, 0)
END SUB

SUB cmdRight_Click ()
    CALL paint_region(1, 0)
END SUB

SUB cmdSort_Click ()
    DIM i AS INTEGER, j AS INTEGER, m AS INTEGER, n AS INTEGER
    DIM current_region AS INTEGER, start_number_of_sectors AS INTEGER
    DIM previous_value AS INTEGER

    FOR i = 0 TO number_of_segments - 1
        FOR j = 1 TO number_of_radii
            IF (region_index(j, i) <= 0) THEN
                MSGBOX "Error: Every sector must be part of a region"
                frmRegion.cmdSort.Tag = "F"
                x_position = 0
                y_position = 0
                IF (number_of_radii < 7) THEN
                    CALL paint_region(0, i)
                    frmRegion.txtSector(j - 1).SETFOCUS
                ELSEIF ((j + 6) <= number_of_radii) THEN
                    CALL paint_region((j - 1), i)
                    frmRegion.txtSector(0).SETFOCUS
                ELSE CALL paint_region((number_of_radii - 7), i)
                    WHILE j > 7
                        j = j - 7
                    WEND
                    frmRegion.txtSector(j - 1).SETFOCUS
                END IF
                EXIT SUB
            END IF
        NEXT j
    NEXT i
    frmRegion.cmdSort.Tag = "Y"

    current_region = 2

    FOR i = 0 TO number_of_segments - 1
        FOR j = 2 TO number_of_radii
            IF (region_index(j, i) >= current_region) THEN
                previous_value = region_index(j, i)
                FOR m = i TO number_of_segments - 1
                    FOR n = 2 TO number_of_radii
                        IF (region_index(n, m) = previous_value) THEN
                            region_index(n, m) = current_region
                        ELSEIF (region_index(n, m) = current_region) THEN
                            region_index(n, m) = 100 + current_region
                        END IF
                    NEXT n
                NEXT m
                current_region = current_region + 1
            END IF
        NEXT j
    NEXT i

```

```

number_econ_regions = current_region - 1
frmProblem_data.lblNumber_of_regions.caption = STR$(number_econ_regions)
CALL paint_region(0, 0)

END SUB

SUB cmdUp_Click ()
    CALL paint_region(0, -1)
END SUB

SUB txtSector_LostFocus (index AS INTEGER)

    DIM i AS INTEGER, j AS INTEGER, k AS INTEGER
    DIM y AS INTEGER

    k = 0
    i = VAL(frmRegion.txtSector(index).text)
    IF (i <> 0) AND (i > 99 OR i < 2) THEN
        MSGBOX "Error: region numbers must be between 2 and 99."
        frmRegion.txtSector(index).SETFOCUS
        EXIT SUB
    END IF

    FOR i = 0 TO 6

        IF (((i + y_position) > 15) AND (y_position > 9)) THEN
            y = (i + y_position - 16)
        ELSEIF ((y_position < 0) AND ((y_position + i) < 0)) THEN
            y = (i + y_position + 16)
        ELSE
            y = (i + y_position)
        END IF

        FOR j = (x_position + 1) TO (x_position + 7)
            IF k = index THEN
                region_index(j, y) = VAL(frmRegion.txtSector(index).text)
                EXIT SUB
            ELSE k = k + 1
            END IF
        NEXT j
    NEXT i
    MSGBOX "Unkown error!"

END SUB

```

**FILENAME: ROSETTE.BAS**

```
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

SUB print_graphic (text AS STRING, text_width AS INTEGER, centered AS INTEGER)
    'This procedure allows the user to output text to the screen in a specified
    'position when forms are not showing.

    DIM next_line AS STRING, character AS STRING
    DIM character_location AS INTEGER, character_found AS INTEGER

    'Break the text into line sized pieces based upon length and upon
    'special characters such as -, <cr> and " ".

    WHILE (LEN(text) > text_width)
        next_line = LEFT$(text, text_width)
        character_found = FALSE
        character_location = text_width
        WHILE ((NOT character_found) AND (character_location > 0))
            character = MID$(next_line, character_location, 1)
            IF ((character = " ") OR (character = "-") OR (character = CHR$(13)))
                THEN
                    character_found = TRUE
                ELSE
                    character_location = character_location - 1
                END IF
            WEND
            IF character_found THEN
                IF character = "-" THEN
                    'If text needs to be centered then center it.
                    IF centered THEN
                        PRINT TAB((text_width - LEN(LEFT$(next_line,
character_location))) / 2);
                    END IF
                    'Print text.
                    PRINT LEFT$(next_line, character_location)
                ELSE
                    'Center next line if necessary.
                    IF centered THEN
                        PRINT TAB((text_width - LEN(LEFT$(next_line, character_location
- 1))) / 2);
                    END IF
                    PRINT LEFT$(next_line, character_location - 1)
                END IF
                character_found = FALSE
                text = MID$(text, character_location + 1)
            ELSE
                PRINT next_line
                text = MID$(text, text_width + 1)
            END IF
        WEND
        IF centered THEN
            PRINT TAB((text_width - LEN(text)) / 2);
        END IF
        PRINT text;
    END SUB

    SUB rosette ()

```

```

DIM paint_value AS INTEGER, max_pop AS LONG, pop_scale AS LONG
DIM scale_interval AS LONG, i AS INTEGER, j AS INTEGER, title AS STRING
DIM max_radius AS SINGLE, rotation_increment AS SINGLE, rotation AS SINGLE
DIM y2 AS SINGLE, y1 AS SINGLE, x2 AS SINGLE, x1 AS SINGLE, r1 AS SINGLE

'Declare constants to determine if text is centered or not and pi.

CONST pi = 3.14159265358979#
CONST paint_position = .75
CONST not_centered = FALSE
CONST centered = TRUE

UNLOAD frmDisclaimer

'Set max pop to the highest value in the population grid.

max_pop = 0
FOR i = 1 TO number_of_segments STEP 1
    FOR j = 1 TO number_of_radii STEP 1
        IF max_pop < sector_population(i, j) THEN
            max_pop = sector_population(i, j)
        END IF
    NEXT j
NEXT i

'Determine which scale to used based upon max pop, and set max pop
'to that scale.

IF max_pop <= 15 THEN
    scale_interval = 1
    max_pop = 15
ELSEIF max_pop <= 150 THEN
    scale_interval = 10
    max_pop = 150
ELSEIF max_pop <= 1500 THEN
    scale_interval = 100
    max_pop = 1500
ELSE
    scale_interval = 7.5 * (10 ^ (INT(LOG(max_pop) / LOG(10)) - 1))
    max_pop = ((max_pop \ scale_interval) + 1) * scale_interval
    scale_interval = max_pop \ 15
END IF

'Set the outermost radius (max radius).

max_radius = radial_distance(number_of_radii)

'Set screen to mode 12 - vga or mcga screen type.

SCREEN.HIDE
SCREEN 12

'Set up and initialize color palate.

PALETTE 15, 0
i = 14
FOR j = 21 TO 63 STEP 21
    PALETTE i, 256 * j + 63
    i = i - 1
NEXT j
FOR j = 42 TO 0 STEP -21
    PALETTE i, 256 * 63 + j

```

```

    i = i - 1
NEXT J
FOR j = 21 TO 63 STEP 21
    PALETTE i, 65536 * j + 256 * 63
    i = i - 1
NEXT J
FOR j = 42 TO 0 STEP -21
    PALETTE i, 65536 * 63 + 256 * j
    i = i - 1
NEXT J
FOR j = 21 TO 42 STEP 21
    PALETTE i, 65536 * 63 + j
    i = i - 1
NEXT J
PALETTE i, 4144959
COLOR 15

'Print standard text on the screen.

title = "SECPOP90 V2.3"
LOCATE 1, 1
CALL print_graphic(title, 80, centered)
title = frmSite_Data.txtSite_Name.text
LOCATE 2, 1
CALL print_graphic(title, 80, centered)
LOCATE 30, 1
CALL print_graphic("Press the space bar to continue.", 80, centered)

'If user has entered valid input, create graph.

IF max_radius <> 0 THEN
    IF max_pop <> 0 THEN
        'Paint color bar next to circle.

        FOR i = 0 TO 14
            LINE (617, 390 - (i * 16))-(637, 374 - (i * 16)), i, BF
            LINE (617, 390 - (i * 16))-(637, 374 - (i * 16)), 15, B
        NEXT i

        'Display the radii in columns, below population scale and units.

        FOR i = 0 TO 15
            pop_scale = scale_interval * i
            LOCATE 25 - i, 67
            CALL print_graphic(LEFT$(RIGHT$(
                "#,##,##,##;0"), 11), 11, not_centered)
        NEXT i

        LOCATE 7, 70
        CALL print_graphic("Population", 10, not_centered)
        LOCATE 8, 70
        CALL print_graphic("    Scale", 8, not_centered)

    END IF

    LOCATE 3, 1
    CALL print_graphic("Radial Distances", 16, not_centered)
    LOCATE 4, 1

    IF frmProblem_Data.optmiles.value = TRUE THEN
        title = "    In Miles"

```

```

        CALL print_graphic(title, 11, not_centered)
ELSE title = "In Kilometers"
        CALL print_graphic(title, 13, not_centered)
END IF

FOR i = 1 TO number_of_radii STEP 2
    LOCATE 4 + (i + 1) / 2, 1
    CALL print_graphic(RIGHT$("      " + FORMAT$(radial_distance(i), ".0"),
6), 6, not_centered)
    IF i < number_of_radii THEN
        LOCATE 4 + (i + 1) / 2, 7
        CALL print_graphic(RIGHT$("      " + FORMAT$(radial_distance(i + 1),
".0"), 6), 6, not_centered)
    END IF
NEXT i

'Set up graphics view port, and coordinate system for it.

VIEW (130, 50)-(510, 430)
y2 = max_radius
y1 = -max_radius
x2 = max_radius
x1 = -max_radius
WINDOW (x1, y2)-(x2, y1)

'Determine the number of degrees per segment.

rotation_increment = 2 * pi / number_of_segments

'Divide the circle up into segments and label segments.

FOR i = 1 TO number_of_segments STEP 1
    rotation = (i - 1) * rotation_increment
    x1 = SIN(rotation) * max_radius
    y1 = COS(rotation) * max_radius
    x2 = PMAP(x1, 0) + (SIN(rotation) * 20)
    y2 = PMAP(y1, 1) - (COS(rotation) * 15)
    LOCATE ((y2 + 50') / 16') + .5, ((x2 + 130') \ 8') + .5
    CALL print_graphic(directions(i) + " ", len(directions(i)) + 1,
not_centered)
    rotation = rotation + (rotation_increment / 2)
    x1 = SIN(rotation) * max_radius
    y1 = COS(rotation) * max_radius
    LINE (0, 0)-(x1, y1)
NEXT i

'Draw circles to signify the different radii.

FOR i = 1 TO number_of_radii STEP 1
    CIRCLE (0', 0'), radial_distance(i)
NEXT i

'Color all of the various circles.

FOR i = 1 TO number_of_segments STEP 1
    rotation = ((i - 1) * rotation_increment)
    FOR j = 1 TO number_of_radii STEP 1
        r1 = (1 - paint_position) * radial_distance(j - 1)
        r1 = r1 + paint_position * radial_distance(j)
        x1 = SIN(rotation) * r1
        y1 = COS(rotation) * r1
        IF max_pop = 0 THEN
            paint_value = 0

```

```
    ELSE
        paint_value = FIX(sector_population(i, j) / max_pop * 15)
        IF paint_value = 15 THEN
            paint_value = 14
        END IF
    END IF
    PAINT (x1, y1), paint_value, 15
NEXT j
NEXT i

END IF

'Wait until space bar is pressed.

WHILE INKEY$ <> " "
WEND

'Reset screen to default values, and show forms.

SCREEN 0
WIDTH 80
SCREEN.SHOW

END SUB
```

**FILENAME: RPTABLE1.BAS**

```
option explicit

'$include: 'secpop90.bi'

'This function calculates and displays all of the fields for table one.

sub paint_table ()

    dim i as integer, j as integer, k as integer
    dim sum as long

    'These two constants are the number of radii and sectors displayed.

    const max_radii_displayed = 4
    const max_sectors_displayed = 6

    'Set the table caption to reflect the table type.

    if (table_type = 1) then
        frmTable_1.Caption = "Population Table"
    else
        frmTable_1.Caption =
            "Cumulative (By Direction) Population Table"
    end if

    'Determine the unit of measure.

    !!!!!!

    if (frmProblem_Data.optKilometers.value) then
        frmTable_1.lblDistance_Heading.caption = _
            "Radii (Kilometers)"
    else
        frmTable_1.lblDistance_Heading.caption = "Radii (Miles)"
    end if

    'Verify that coordinates are within array.

    if (x_position < 1) then
        x_position = 1
    elseif (number_of_radii <= max_radii_displayed) then
        x_position = 1
    elseif ((x_position + max_radii_displayed - 1) > _
        number_of_radii) then
        x_position = number_of_radii - max_radii_displayed + 1
    end if

    if (y_position < 1) then
        y_position = 1
    elseif ((y_position + max_sectors_displayed + 1) > _
        number_of_segments) then
```

```

    y_position = number_of_segments - max_sectors_displayed + 1
end if

'Fill the radii label values.

for i = 0 to max_radii_displayed - 1 step 1

    if ((x_position + i) <= number_of_radii) then
        frmTable_1.lblDistance_array(i).caption =
            format$(radial_distance(x_position + i), "###0.0000")
    else
        frmTable_1.lblDistance_array(i).caption = ""
    end if

next i

'Fill the direction label values.

for i = 0 to max_sectors_displayed - 1 step 1

    frmTable_1.lblSector_array(i).caption = _
        directions(y_position + i)

next i

` 

'Fill the population sector label values with either the population
'data or the cumulative population data depending on which table
'type we're displaying. Also calculate and fill the sector sum
'labels.

for i = 0 to max_sectors_displayed - 1 step 1

    sum = 0

    frmTable_1.lblValue_array(i).caption = ""

    for j = 1 to number_of_radii step 1

        sum = sum + sector_population(y_position + i, j)

        if ((j >= x_position) and
            (j <= (x_position + max_radii_displayed - 1))) then

            if (table_type = 1) then

                frmTable_1.lblValue_array(i).caption =
                    frmTable_1.lblvalue_array(i).caption + -
                    right$(format$(sector_population(
                        y_position + i, j), "#####0"), 10)

            else

                frmTable_1.lblValue_array(i).caption =
                    frmTable_1.lblvalue_array(i).caption + -
                    right$(format$(sum, "#####0"), 10)

            end if

        end if

    end if

```

```

next j

frmTable_1.lblCsum_array(i).caption = format$(sum, "#####0")

next i

'Calculate and fill the radial sum labels.

for i = 0 to max_radius_displayed - 1 step 1

  if ((x_position + i) <= number_of_radius) then

    sum = 0

    if (table_type = 1) then
      for j = 1 to number_of_segments step 1
        sum = sum + sector_population(j, x_position + i)
      next j
    else
      for j = 1 to number_of_segments step 1
        for k = 1 to (x_position + i) step 1
          sum = sum + sector_population(j, k)
        next k
      next j
    end if

    frmTable_1.lblRsum_array(i).caption = _
      format$(sum, "#####0")

  else

    frmTable_1.lblRsum_array(i).caption = ""

  end if

next i

'Calculate and fill the total sum label.

sum = 0

for i = 1 to number_of_segments step 1

  for j = 1 to number_of_radius step 1

    sum = sum + sector_population(i, j)

  next j

next i

frmTable_1.lblTotal.caption = format$(sum, "#####0")

end sub

```

**FILENAME: SAVEDATA.BAS**

```
OPTION EXPLICIT

SUB save_data ()
    DIM filenum AS INTEGER
    DIM ForeColor AS INTEGER, BackColor AS INTEGER
    DIM Flags AS INTEGER, Cancel AS INTEGER
    DIM filename AS STRING, PathName AS STRING
    DIM DefaultExt AS STRING, DialogTitle AS STRING
    DefaultExt = "*.DAT"
    DialogTitle = "Save Data"

    IF frmProblem_Data.txtRadial_Distance1.text = "" THEN
        MSGBOX "Data is not yet available - Site file has not been loaded"
        EXIT SUB
    END IF

    CALL FileSave(filename, PathName, DefaultExt, DialogTitle, 7, 1, Flags, Cancel)
    IF NOT Cancel THEN
        IF PathName <> "" THEN filename = PathName + "\" + filename

        ON LOCAL ERROR GOTO errfix
        frmSaving.SHOW
        PRINTER.PrintTarget = filename
        CALL print_MACCS_input_file
        PRINTER.PrintTarget = ""
        CLOSE filenum
        UNLOAD frmSaving
    END IF
    EXIT SUB

errfix:
    MSGBOX "A FILE ERROR HAS OCCURRED. PLEASE CHECK ALL FILE NAMES"
    EXIT SUB
END SUB
```

**FILENAME: SECPOP90.BAS**

OPTION EXPLICIT

'\$INCLUDE: 'secpop90.bi'

'Define all of the dynamic arrays used in SECPOP90.

REDIM SHARED radial\_distance(max\_number\_of\_radii) AS SINGLE  
REDIM SHARED raddis(max\_number\_of\_radii) AS SINGLE  
REDIM SHARED regional\_radii(max\_number\_of\_radii) AS SINGLE  
REDIM SHARED population(max\_number\_of\_radii) as long  
REDIM SHARED population\_threshold(max\_number\_of\_radii) as long

REDIM SHARED region\_index(max\_number\_of\_radii, \_  
 number\_of\_segments) AS INTEGER  
REDIM SHARED econ\_data(max\_econ\_regions) AS economic\_data

REDIM SHARED sector\_population(number\_of\_segments, \_  
 max\_number\_of\_radii) AS LONG  
REDIM SHARED sector\_area(number\_of\_segments, \_  
 max\_number\_of\_radii) AS LONG  
REDIM SHARED sector\_frcrnd(number\_of\_segments, \_  
 max\_number\_of\_radii) AS SINGLE

REDIM SHARED county\_state(number\_of\_counties) AS STRING \* 2  
REDIM SHARED county\_name(number\_of\_counties) \_  
 AS STRING \* max\_county\_name\_length  
REDIM SHARED county\_frcrnd(number\_of\_counties) AS SINGLE  
REDIM SHARED county\_frmfrc(number\_of\_counties) AS SINGLE  
REDIM SHARED county\_dpf(number\_of\_counties) AS SINGLE  
REDIM SHARED county\_asfp(number\_of\_counties) AS SINGLE  
REDIM SHARED county\_vfrm(number\_of\_counties) AS SINGLE  
REDIM SHARED county\_vnfrm(number\_of\_counties) AS SINGLE

REDIM SHARED directions(number\_of\_segments) AS STRING

REDIM SHARED layer\_number(max\_layers) AS INTEGER  
REDIM SHARED data\_flag(max\_layers) AS INTEGER  
REDIM SHARED display\_flag(max\_layers) AS INTEGER  
REDIM SHARED active\_layer(max\_layers) AS INTEGER  
REDIM SHARED layer\_name(max\_layers) AS STRING  
REDIM SHARED pts\_color(max\_layers) AS INTEGER  
REDIM SHARED pts\_type(max\_layers) AS INTEGER  
REDIM SHARED pts\_size(max\_layers) AS INTEGER  
REDIM SHARED pts\_mode(max\_layers) AS INTEGER  
REDIM SHARED lines\_color(max\_layers) AS INTEGER  
REDIM SHARED lines\_type(max\_layers) AS INTEGER  
REDIM SHARED lines\_size(max\_layers) AS INTEGER  
REDIM SHARED lines\_mode(max\_layers) AS INTEGER  
REDIM SHARED polyg\_color(max\_layers) AS INTEGER  
REDIM SHARED polyg\_type(max\_layers) AS INTEGER  
REDIM SHARED polyg\_size(max\_layers) AS INTEGER  
REDIM SHARED polyg\_mode(max\_layers) AS INTEGER  
REDIM SHARED text\_color(max\_layers) AS INTEGER  
REDIM SHARED text\_type(max\_layers) AS INTEGER  
REDIM SHARED text\_size(max\_layers) AS INTEGER  
REDIM SHARED text\_mode(max\_layers) AS INTEGER

REDIM SHARED radial\_area(max\_number\_of\_radii) AS SINGLE

'Load and show the main SECPOP90 form.

```
LOAD frmMain
frmMain.SHOW
```

**FILENAME: SETUP.FRM**

```
Version 1.00
BEGIN Form frmSetup
    AutoRedraw      = 0
    BackColor       = QBColor(3)
    BorderStyle     = 1
    Caption         = "Setup"
    ControlBox      = -1
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(20)
    Left             = Char(0)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag              = ""
    Top              = Char(1)
    Visible         = -1
    Width            = Char(78)
    WindowState     = 0
    BEGIN Frame fraOutput
        BackColor      = QBColor(3)
        Caption         = ""
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(3)
        Left             = Char(39)
        MousePointer    = 0
        TabIndex        = 14
        Tag              = ""
        Top              = Char(11)
        Visible         = -1
        Width            = Char(35)
    END
    BEGIN Label lblSave_type
        Alignment       = 0
        AutoSize        = -1
        BackColor       = QBColor(3)
        BorderStyle     = 0
        Caption         = "Site-Specific Output File Format"
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(1)
        Left             = Char(3)
        MousePointer    = 0
        TabIndex        = 13
        Tag              = ""
        Top              = Char(12)
        Visible         = -1
        Width            = Char(32)
    END
    BEGIN Label lblProb_path
        Alignment       = 0
        AutoSize        = -1
        BackColor       = QBColor(3)
        BorderStyle     = 0
        Caption         = "Default &Problem File Path"
        DragMode        = 0
        Enabled         = -1
    END
```

```

ForeColor      = QBColor(0)
Height        = Char(1)
Left          = Char(40)
MousePointer  = 0
TabIndex      = 2
Tag           = ""
Top           = Char(1)
Visible       = -1
Width         = Char(26)

END
BEGIN Label lblOutput_Path
    Alignment    = 0
    AutoSize     = -1
    BackColor    = QBColor(3)
    BorderStyle  = 0
    Caption      = "Default &Output File Path"
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(1)
    Left         = Char(3)
    MousePointer = 0
   TabIndex      = 4
    Tag          = ""
    Top          = Char(6)
    Visible      = -1
    Width        = Char(25)
END
BEGIN Label lblData_Path
    Alignment    = 0
    AutoSize     = -1
    BackColor    = QBColor(3)
    BorderStyle  = 0
    Caption      = "&Location of Census Databases"
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(1)
    Left         = Char(40)
    MousePointer = 0
   TabIndex      = 6
    Tag          = ""
    Top          = Char(6)
    Visible      = -1
    Width        = Char(29)
END
BEGIN Label lblsite_path
    Alignment    = 0
    AutoSize     = -1
    BackColor    = QBColor(3)
    BorderStyle  = 0
    Caption      = "&Default Site File Path"
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(1)
    Left         = Char(3)
    MousePointer = 0
   TabIndex      = 0
    Tag          = ""
    Top          = Char(1)
    Visible      = -1
    Width        = Char(23)

```

```

END
BEGIN TextBox txtSite_Path
    BackColor      = QBColor(3)
    BorderStyle   = 1
    DragMode      = 0
    Enabled       = -1
    ForeColor     = QBColor(0)
    Height        = Char(3)
    Left          = Char(2)
    MousePointer = 0
    MultiLine     = 0
    ScrollBars   = 0
    TabIndex      = 1
    TabStop       = -1
    Tag           = ""
    Text          = "SITES"
    Top           = Char(2)
    Visible      = -1
    Width         = Char(35)
END
BEGIN TextBox txtProblem_path
    BackColor      = QBColor(3)
    BorderStyle   = 1
    DragMode      = 0
    Enabled       = -1
    ForeColor     = QBColor(0)
    Height        = Char(3)
    Left          = Char(39)
    MousePointer = 0
    MultiLine     = 0
    ScrollBars   = 0
    TabIndex      = 3
    TabStop       = -1
    Tag           = ""
    Text          = "SITES"
    Top           = Char(2)
    Visible      = -1
    Width         = Char(35)
END
BEGIN TextBox txtOutput_path
    BackColor      = QBColor(3)
    BorderStyle   = 1
    DragMode      = 0
    Enabled       = -1
    ForeColor     = QBColor(0)
    Height        = Char(3)
    Left          = Char(2)
    MousePointer = 0
    MultiLine     = 0
    ScrollBars   = 0
    TabIndex      = 5
    TabStop       = -1
    Tag           = ""
    Text          = "."
    Top           = Char(7)
    Visible      = -1
    Width         = Char(35)
END
BEGIN TextBox txtData_path
    BackColor      = QBColor(3)
    BorderStyle   = 1
    DragMode      = 0
    Enabled       = -1

```

```

ForeColor      = QBColor(0)
Height        = Char(3)
Left          = Char(39)
MousePointer  = 0
MultiLine     = 0
ScrollBars    = 0
TabIndex      = 7
TabStop       = -1
Tag           = ""
Text          = "CENSUS"
Top           = Char(7)
Visible       = -1
Width         = Char(35)

END
BEGIN OptionButton optCSV
    BackColor    = QBColor(3)
    Caption      = "CS&V"
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(1)
    Left         = Char(60)
    MousePointer = 0
    TabIndex     = 9
    TabStop      = 0
    Tag          = ""
    Top          = Char(12)
    Value        = 0
    Visible      = -1
    Width        = Char(7)

END
BEGIN OptionButton optMACCS
    BackColor    = QBColor(3)
    Caption      = "&MACCS"
    DragMode     = 0
    Enabled      = -1
    ForeColor    = QBColor(0)
    Height       = Char(1)
    Left         = Char(45)
    MousePointer = 0
    TabIndex     = 8
    TabStop      = -1
    Tag          = ""
    Top          = Char(12)
    Value        = -1
    Visible      = -1
    Width        = Char(9)

END
BEGIN CommandButton cmdMapplan_Setup
    BackColor    = QBColor(3)
    Cancel       = 0
    Caption      = "M&apPlan"
    Default      = 0
    DragMode     = 0
    Enabled      = -1
    Height       = Char(3)
    Left         = Char(8)
    MousePointer = 0
    TabIndex     = 10
    TabStop      = -1
    Tag          = ""
    Top          = Char(15)
    Visible      = -1

```

```

        Width      = Char(16)
END
BEGIN CommandButton cmdSave_Settings
    BackColor  = QBColor(3)
    Cancel     = 0
    Caption    = "&Save Changes"
    Default    = 0
    DragMode   = 0
    Enabled    = -1
    Height     = Char(3)
    Left       = Char(29)
    MousePointer = 0
    TabIndex   = 11
    TabStop    = -1
    Tag        = ""
    Top        = Char(15)
    Visible   = -1
    Width      = Char(16)
END
BEGIN CommandButton cmdCancel
    BackColor  = QBColor(3)
    Cancel     = -1
    Caption    = "&Close"
    Default    = -1
    DragMode   = 0
    Enabled    = -1
    Height     = Char(3)
    Left       = Char(51)
    MousePointer = 0
    TabIndex   = 12
    TabStop    = -1
    Tag        = ""
    Top        = Char(15)
    Visible   = -1
    Width      = Char(16)
END
END
OPTION EXPLICIT

```

'\$INCLUDE: 'secpop90.bi'

'This form allows the user to set certain paths which will be remembered  
 'by the program until they are changed. These paths include input and  
 'output directories, among other things. Setup also allows the user to  
 'specify which file format they would like to use with their output  
 'data files - MACCS input file or comma seperated variable formats.

SUB cmdCancel\_Click ()

'If the user selects the cancel button from setup, the setup form is  
 'hidden, but the changes are retained until the program is exited.

frmSetup.HIDE

END SUB

SUB cmdMapplan\_Setup\_Click ()

RUN "mp\_setup.exe"

END SUB

SUB cmdSave\_Settings\_Click ()

```

'This routine saves setup data to a file so that the program has
'access to it every time the program is run.

DIM filenum AS INTEGER

'Locate first free file handle.

filenum = FREEFILE

'Open file (called secpop90.cfg) and output all of the data to it.
OPEN "SECPOP90.CFG" FOR OUTPUT AS #filenum
PRINT #filenum, frmSetup.txtSite_path.text
PRINT #filenum, frmSetup.txtProblem_path.text
PRINT #filenum, frmSetup.txtOutput_path.text
PRINT #filenum, frmSetup.txtData_path.text
IF frmSetup.optMACCS.value = -1 THEN PRINT #filenum, "MACCS" ELSE PRINT
#filenum, "CSV"
CLOSE #filenum

END SUB

SUB txtData_path_KeyPress (KeyAscii AS INTEGER)

'Convert all input to upper case.

DIM c AS STRING

c = CHR$(KeyAscii)

KeyAscii = ASC(UCASE$(c$))

END SUB

SUB txtOutput_path_KeyPress (KeyAscii AS INTEGER)

'Convert all input to upper case.

DIM c AS STRING

c = CHR$(KeyAscii)

KeyAscii = ASC(UCASE$(c$))

END SUB

SUB txtProblem_path_KeyPress (KeyAscii AS INTEGER)

'Convert all input to upper case.

DIM c AS STRING

c = CHR$(KeyAscii)

KeyAscii = ASC(UCASE$(c$))

END SUB

SUB txtSite_Path_KeyPress (KeyAscii AS INTEGER)

'Convert all input to upper case.

DIM c AS STRING

```

```
c = CHR$(KeyAscii)  
KeyAscii = ASC(UCASE$(c$))  
END SUB
```

~

**FILENAME: SITECODE.BAS**

```
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

'This module allows the user to create, edit, and and view site files

SUB new_site ()

    'This module initializes the new site form to its default values,
    'and shows the form.

    frmMain.mnuSave_SiteCom.Enabled = FALSE
    frmMain.mnuSave_As_SiteCom.Enabled = FALSE
    frmSite_Data.Caption = CURDIR$ + "\\" + "NEW_SITE.SIT"
    frmProblem_Data.txtSite_File_Name.text = "NEW_SITE.SIT"
    frmSite_Data.txtSite_Name.text = ""
    frmSite_Data.txtLongitude_Degrees.text = ""
    frmSite_Data.txtLongitude_Minutes.text = ""
    frmSite_Data.txtLongitude_Seconds.text = ""
    frmSite_Data.txtLatitude_Degrees.text = ""
    frmSite_Data.txtLatitude_Minutes.text = ""
    frmSite_Data.txtLatitude_Seconds.text = ""
    frmSite_Data.txtSite_Remarks.text = ""
    frmSite_Data.SHOW MODAL

END SUB

SUB open_site ()

    'This routine opens a previously created site file.

    DIM site_name AS STRING, site_remarks AS STRING, dummy AS STRING
    DIM longitude_degrees AS INTEGER, longitude_minutes AS INTEGER,
    longitude_seconds AS INTEGER
    DIM latitude_degrees AS INTEGER, latitude_minutes AS INTEGER, latitude_seconds
    AS INTEGER
    DIM ForeColor AS INTEGER, BackColor AS INTEGER
    DIM Flags AS INTEGER, Cancel AS INTEGER
    DIM filename AS STRING
    STATIC Pathname AS STRING
    DIM DefaultExt AS STRING, DialogTitle AS STRING
    CONST site_file = 20, file_buffer = 1024

    'Initialize file dialog box.

    DefaultExt = "*.SIT"
    Pathname = frmSetup.txtSite_path.text
    DialogTitle = "Open Site"
    BackColor = WHITE
    ForeColor = BLACK

    'Enable local error checking.

    ON LOCAL ERROR GOTO site_file_error

    'Call file dialogue box.

    CALL FileOpen(filename, Pathname, DefaultExt, DialogTitle, ForeColor, BackColor,
    Flags, Cancel)
```

```

'If user did not select cancel from the file dialog box then proceed.

IF NOT Cancel THEN

    'If file is not in current directory then add path to filename.

    IF Pathname <> "" THEN filename = Pathname + "\" + filename

    'Open site file and read in the site data, placing it into temp variables.

    OPEN filename FOR INPUT ACCESS READ LOCK WRITE AS #site_file LEN =
file_buffer
        LINE INPUT #site_file, dummy
        LINE INPUT #site_file, site_name
        LINE INPUT #site_file, dummy
        LINE INPUT #site_file, dummy
        INPUT #site_file, longitude_degrees
        INPUT #site_file, longitude_minutes
        INPUT #site_file, longitude_seconds
        LINE INPUT #site_file, dummy
        INPUT #site_file, latitude_degrees
        INPUT #site_file, latitude_minutes
        INPUT #site_file, latitude_seconds
        LINE INPUT #site_file, dummy
        LINE INPUT #site_file, site_remarks
    CLOSE site_file

    'Set window caption to filename.

    frmSite_Data.Caption = filename

    'Move file values from temporary variables to the correct form locations.

    frmProblem_Data.txtSite_File_Name.text = filename
    frmSite_Data.txtSite_Name.text = site_name
    frmSite_Data.txtLongitude_Degrees.text = STR$(longitude_degrees)
    frmSite_Data.txtLongitude_Minutes.text = STR$(longitude_minutes)
    frmSite_Data.txtLongitude_Seconds.text = STR$(longitude_seconds)
    frmSite_Data.txtLatitude_Degrees.text = STR$(latitude_degrees)
    frmSite_Data.txtLatitude_Minutes.text = STR$(latitude_minutes)
    frmSite_Data.txtLatitude_Seconds.text = STR$(latitude_seconds)
    frmSite_Data.txtSite_Remarks.text = site_remarks

    'If site form is not visible, show it.

    IF (frmSite_Data.Visible = FALSE) THEN
        frmSite_Data.SHOW MODAL
    END IF

    END IF

    EXIT SUB

site_file_error:

    'If an error occurs, inform the user and exit the sub routine.

    MSGBOX "Error in opening or reading site file - check file name."
    EXIT SUB

END SUB

SUB print_site ()

```

```

'Print data from a site file.

DIM ForeColor AS INTEGER, BackColor AS INTEGER
DIM Copies AS INTEGER, Cancel AS INTEGER, i AS INTEGER
DIM site_name AS STRING, site_remarks AS STRING

'Declare as constants the print margins.

CONST left_margin = 10
CONST text_width = 40

BackColor = WHITE
ForeColor = BLACK

'Enable local error correction.

ON LOCAL ERROR GOTO site_print_error

'Open the printer dialog box.

CALL FilePrint(Copies, ForeColor, BackColor, Cancel)

'If the user did not select cancel from the printer dialog then proceed.

IF NOT Cancel THEN

    'Execute loop once for each copy the user has requested.

    FOR i = 1 TO Copies
        PRINTER.PRINT TAB(left_margin); "File Name:";
        PRINTER.PRINT TAB(left_margin + LEN("Site Coordinates:   "));
        frmSite_Data.Caption
        PRINTER.PRINT
        PRINTER.PRINT TAB(left_margin); "Site Name:";
        site_name = frmSite_Data.txtSite_Name.text
        CALL print_text(site_name, text_width, left_margin + LEN("Site
Coordinates:   "));
        PRINTER.PRINT
        PRINTER.PRINT TAB(left_margin); "Site Coordinates:";
        PRINTER.PRINT TAB(left_margin + LEN("Site Coordinates:   ")); "Longitude
(Degrees, Minutes, Seconds)"
        PRINTER.PRINT USING "           #####"; TAB(left_margin + LEN("Site
Coordinates:   ")); VAL(frmSite_Data.txtLongitude_Degrees.text);
        PRINTER.PRINT USING "           ##";
        VAL(frmSite_Data.txtLongitude_Minutes.text);
        PRINTER.PRINT USING "           ##";
        VAL(frmSite_Data.txtLongitude_Seconds.text)
        PRINTER.PRINT TAB(left_margin + LEN("Site Coordinates:   ")); "Latitude
(Degrees, Minutes, Seconds)"
        PRINTER.PRINT USING "           #####"; TAB(left_margin + LEN("Site
Coordinates:   ")); VAL(frmSite_Data.txtLatitude_Degrees.text);
        PRINTER.PRINT USING "           ##";
        VAL(frmSite_Data.txtLatitude_Minutes.text);
        PRINTER.PRINT USING "           ##";
        VAL(frmSite_Data.txtLatitude_Seconds.text)
        PRINTER.PRINT
        PRINTER.PRINT TAB(left_margin); "Site Remarks:";
        site_remarks = frmSite_Data.txtSite_Remarks.text
        CALL print_text(site_remarks, text_width, left_margin + LEN("Site
Coordinates:   "));
        PRINTER.NEWPAGE
    NEXT i

```

```

PRINTER.ENDDOC

END IF

EXIT SUB

site_print_error:
    'IF an error occurs, notify the user and exit the sub routine.

    MSGBOX "Device error - Check printer or other print device."
    EXIT SUB

END SUB

SUB save_as_site ()
    'Save a site file using a new name.

    CONST site_file = 20, file_buffer = 1024

    DIM site_name AS STRING, site_remarks AS STRING, dummy AS STRING, Caption AS
    STRING
        DIM longitude_degrees AS INTEGER, longitude_minutes AS INTEGER,
    longitude_seconds AS INTEGER
        DIM latitude_degrees AS INTEGER, latitude_minutes AS INTEGER, latitude_seconds
    AS INTEGER
        DIM ForeColor AS INTEGER, BackColor AS INTEGER
        DIM Flags AS INTEGER, Cancel AS INTEGER
        DIM filename AS STRING, Pathname AS STRING
        DIM DefaultExt AS STRING, DialogTitle AS STRING
        DIM temp AS INTEGER

    'Check validity of data entered by user, warn user if there is a problem.

    temp = verify_input(2)

    'Set default file to name of site window.

    Caption = frmSite_Data.Caption

    'Remove path from filename.

    CALL parse_path_and_file(Caption, Pathname, filename)

    'Initialize file dialog box.

    DefaultExt = "*.SIT"
    Pathname = frmSetup.txtSite_path.text
    DialogTitle = "Save As Site"
    BackColor = WHITE
    ForeColor = BLACK

    'Call file dialogue box.

    CALL FileSave(filename, Pathname, DefaultExt, DialogTitle, ForeColor, BackColor,
Flags, Cancel)

    'If user did not select cancel from the file dialog box, then proceed.

    IF NOT Cancel THEN

        'If file is not in the current directory, then add the path to

```

```

'the filename.

IF Pathname <> "" THEN filename = Pathname + "\" + filename
frmSite_Data.Caption = filename
frmProblem_Data.txtSite_File_Name.text = filename

'Call function to save the data.

CALL save_site

END IF

END SUB

SUB save_site ()

'Save data from a site form.

'Declare as constant the site file handle and the file buffer size.

CONST site_file = 20, file_buffer = 1024

DIM filename AS STRING, Pathname AS STRING
DIM DefaultExt AS STRING, DialogTitle AS STRING

'Set default filename to name of site window.
filename = frmSite_Data.Caption

'Enable local error checking.

ON LOCAL ERROR GOTO site_save_error

'If the site has not yet been named, call save as.

IF filename = "" THEN
    CALL save_as_site
    EXIT SUB
END IF

'Open the site file and send the site info to it.

OPEN filename FOR OUTPUT ACCESS WRITE LOCK READ WRITE AS #site_file LEN =
file_buffer
PRINT #site_file, "Site Name:"
PRINT #site_file, frmSite_Data.txtSite_Name.text
PRINT #site_file, "Site Coordinates:"
PRINT #site_file, "Longitude (Degrees, Minutes, Seconds)"
PRINT #site_file, USING "      ####";
VAL(frmSite_Data.txtLongitude_Degrees.text);
PRINT #site_file, USING "      ##";
VAL(frmSite_Data.txtLongitude_Minutes.text);
PRINT #site_file, USING "      ##"; VAL(frmSite_Data.txtLongitude_Seconds.text)
PRINT #site_file, "Latitude (Degrees, Minutes, Seconds)"
PRINT #site_file, USING "      ####";
VAL(frmSite_Data.txtLatitude_Degrees.text);
PRINT #site_file, USING "      ##"; VAL(frmSite_Data.txtLatitude_Minutes.text);
PRINT #site_file, USING "      ##"; VAL(frmSite_Data.txtLatitude_Seconds.text)
PRINT #site_file, "Site Remarks:"
PRINT #site_file, frmSite_Data.txtSite_Remarks.text
CLOSE site_file

'Reset window names to match site file name.

```

```
frmSite_Data.Caption = filename
frmProblem_Data.txtSite_File_Name.text = filename

EXIT SUB

site_save_error:
    'If an error occurs, inform the user and exit sub.

    MSGBOX "Error opening or creating site file"
    EXIT SUB

END SUB
```

**FILENAME: SITEFORM.FRM**

```
Version 1.00
BEGIN Form frmSite_Data
    AutoRedraw      = 0
    BackColor       = QBColor(3)
    BorderStyle     = 1
    Caption         = ""
    ControlBox      = 0
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(21)
    Left            = Char(1)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag             = ""
    Top             = Char(1)
    Visible         = -1
    Width           = Char(76)
    WindowState     = 0
    BEGIN TextBox txtSite_Name
        BackColor      = QBColor(3)
        BorderStyle    = 1
        DragMode       = 0
        Enabled        = -1
        ForeColor      = QBColor(0)
        Height          = Char(4)
        Left            = Char(15)
        MousePointer   = 0
        MultiLine      = -1
        ScrollBars     = 0
        TabIndex        = 1
        TabStop         = -1
        Tag             = ""
        Text            = ""
        Top             = Char(0)
        Visible         = -1
        Width           = Char(43)
    END
    BEGIN Label Label2
        Alignment      = 0
        AutoSize       = 0
        BackColor      = QBColor(3)
        BorderStyle    = 0
        Caption         = "Site Coordinates:"
        DragMode       = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(1)
        Left            = Char(1)
        MousePointer   = 0
        TabIndex        = 2
        Tag             = ""
        Top             = Char(5)
        Visible         = -1
        Width           = Char(18)
    END
    BEGIN TextBox txtLongitude_Degrees
        BackColor      = QBColor(3)
        BorderStyle    = 1
        DragMode       = 0
    END
```

```

        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Left         = Char(48)
        MousePointer = 0
        MultiLine    = 0
        ScrollBars   = 0
        TabIndex     = 14
        TabStop      = -1
        Tag          = ""
        Text          = ""
        Top          = Char(8)
        Visible      = -1
        Width         = Char(7)
    END
BEGIN TextBox txtLongitude_Minutes
        BackColor    = QBColor(3)
        BorderStyle  = 1
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Left         = Char(58)
        MousePointer = 0
        MultiLine    = 0
        ScrollBars   = 0
        TabIndex     = 15
        TabStop      = -1
        Tag          = ""
        Text          = ""
        Top          = Char(8)
        Visible      = -1
        Width         = Char(5)
    END
BEGIN TextBox txtLongitude_Seconds
        BackColor    = QBColor(3)
        BorderStyle  = 1
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Left         = Char(67)
        MousePointer = 0
        MultiLine    = 0
        ScrollBars   = 0
        TabIndex     = 16
        TabStop      = -1
        Tag          = ""
        Text          = ""
        Top          = Char(8)
        Visible      = -1
        Width         = Char(5)
    END
BEGIN TextBox txtLatitude_Degrees
        BackColor    = QBColor(3)
        BorderStyle  = 1
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Left         = Char(13)
        MousePointer = 0
        MultiLine    = 0
    END

```

```

        ScrollBars      = 0
        TabIndex        = 10
        TabStop         = -1
        Tag             = ""
        Text            = ""
        Top             = Char(8)
        Visible         = -1
        Width           = Char(5)

    END
    BEGIN TextBox txtLatitude_Minutes
        BackColor      = QBColor(3)
        BorderStyle    = 1
        DragMode       = 0
        Enabled        = -1
        ForeColor      = QBColor(0)
        Height          = Char(3)
        Left            = Char(22)
        MousePointer   = 0
        MultiLine      = 0
        ScrollBars     = 0
        TabIndex        = 11
        TabStop         = -1
        Tag             = ""
        Text            = ""
        Top             = Char(8)
        Visible         = -1
        Width           = Char(5)

    END
    BEGIN TextBox txtLatitude_Seconds
        BackColor      = QBColor(3)
        BorderStyle    = 1
        DragMode       = 0
        Enabled        = -1
        ForeColor      = QBColor(0)
        Height          = Char(3)
        Left            = Char(31)
        MousePointer   = 0
        MultiLine      = 0
        ScrollBars     = 0
        TabIndex        = 12
        TabStop         = -1
        Tag             = ""
        Text            = ""
        Top             = Char(8)
        Visible         = -1
        Width           = Char(5)

    END
    BEGIN Label Label5
        Alignment       = 0
        AutoSize        = 0
        BackColor      = QBColor(3)
        BorderStyle    = 0
        Caption         = "Degrees"
        DragMode       = 0
        Enabled        = -1
        ForeColor      = QBColor(0)
        Height          = Char(1)
        Left            = Char(47)
        MousePointer   = 0
        TabIndex        = 6
        Tag             = ""
        Top             = Char(7)
        Visible         = -1

```

```

        Width      = Char(7)
END
BEGIN Label Label6
    Alignment  = 0
    AutoSize   = 0
    BackColor  = QBColor(3)
    BorderStyle= 0
    Caption    = "Minutes"
    DragMode   = 0
    Enabled    = -1
    ForeColor  = QBColor(0)
    Height     = Char(1)
    Left       = Char(57)
    MousePointer = 0
    TabIndex   = 7
    Tag        = ""
    Top        = Char(7)
    Visible   = -1
    Width      = Char(7)
END
BEGIN Label Label7
    Alignment  = 0
    AutoSize   = 0
    BackColor  = QBColor(3)
    BorderStyle= 0
    Caption    = "Seconds"
    DragMode   = 0
    Enabled    = -1
    ForeColor  = QBColor(0)
    Height     = Char(1)
    Left       = Char(66)
    MousePointer = 0
    TabIndex   = 8
    Tag        = ""
    Top        = Char(7)
    Visible   = -1
    Width      = Char(7)
END
BEGIN TextBox txtSite_Remarks
    BackColor  = QBColor(3)
    BorderStyle= 1
    DragMode   = 0
    Enabled    = -1
    ForeColor  = QBColor(0)
    Height     = Char(4)
    Left       = Char(15)
    MousePointer = 0
    MultiLine  = -1
    ScrollBars = 2
    TabIndex   = 18
    TabStop    = -1
    Tag        = ""
    Text       = ""
    Top        = Char(12)
    Visible   = -1
    Width      = Char(43)
END
BEGIN Label Label8
    Alignment  = 0
    AutoSize   = 0
    BackColor  = QBColor(3)
    BorderStyle= 0
    Caption    = "Degrees"

```

```

DragMode      = 0
Enabled       = -1
ForeColor     = QBColor(0)
Height        = Char(1)
Left          = Char(12)
MousePointer  = 0
TabIndex      = 3
Tag           = ""
Top           = Char(7)
Visible       = -1
Width         = Char(7)

END
BEGIN Label Label9
    Alignment     = 0
    AutoSize      = 0
    BackColor     = QBColor(3)
    BorderStyle   = 0
    Caption       = "Minutes"
    DragMode      = 0
    Enabled       = -1
    ForeColor     = QBColor(0)
    Height        = Char(1)
    Left          = Char(21)
    MousePointer  = 0
   TabIndex      = 4
    Tag           = ""
    Top           = Char(7)
    Visible       = -1
    Width         = Char(7)

END
BEGIN Label Label10
    Alignment     = 0
    AutoSize      = 0
    BackColor     = QBColor(3)
    BorderStyle   = 0
    Caption       = "Seconds"
    DragMode      = 0
    Enabled       = -1
    ForeColor     = QBColor(0)
    Height        = Char(1)
    Left          = Char(30)
    MousePointer  = 0
   TabIndex      = 5
    Tag           = ""
    Top           = Char(7)
    Visible       = -1
    Width         = Char(7)

END
BEGIN CommandButton cmdOpen
    BackColor     = QBColor(3)
    Cancel        = 0
    Caption       = "S&ites"
    Default       = 0
    DragMode      = 0
    Enabled       = -1
    Height        = Char(3)
    Left          = Char(1)
    MousePointer  = 0
   TabIndex      = 19
    TabStop       = -1
    Tag           = ""
    Top           = Char(16)
    Visible       = -1

```

```

        Width      = Char(11)
END
BEGIN CommandButton cmdSave
    BackColor  = QBColor(3)
    Cancel     = 0
    Caption    = "&Save"
    Default    = 0
    DragMode   = 0
    Enabled    = -1
    Height     = Char(3)
    Left       = Char(31)
    MousePointer = 0
   TabIndex   = 21
    TabStop    = -1
    Tag        = ""
    Top        = Char(16)
    Visible   = -1
    Width      = Char(11)
END
BEGIN CommandButton cmdSave_As
    BackColor  = QBColor(3)
    Cancel     = 0
    Caption    = "Save &As"
    Default    = 0
    DragMode   = 0
    Enabled    = -1
    Height     = Char(3)
    Left       = Char(46)
    MousePointer = 0
   TabIndex   = 22
    TabStop    = -1
    Tag        = ""
    Top        = Char(16)
    Visible   = -1
    Width      = Char(11)
END
BEGIN CommandButton cmdPrint
    BackColor  = QBColor(3)
    Cancel     = 0
    Caption    = "&Print"
    Default    = 0
    DragMode   = 0
    Enabled    = -1
    Height     = Char(3)
    Left       = Char(61)
    MousePointer = 0
   TabIndex   = 23
    TabStop    = -1
    Tag        = ""
    Top        = Char(16)
    Visible   = -1
    Width      = Char(11)
END
BEGIN CommandButton cmdClose
    BackColor  = QBColor(3)
    Cancel     = -1
    Caption    = "&Close"
    Default    = -1
    DragMode   = 0
    Enabled    = -1
    Height     = Char(3)
    Left       = Char(16)
    MousePointer = 0

```

```

TabIndex      = 20
TabStop       = -1
Tag           = ""
Top           = Char(16)
Visible       = -1
Width         = Char(11)

END
BEGIN Label Label1
  Alignment     = 0
  AutoSize      = 0
  BackColor     = QBColor(3)
  BorderStyle   = 0
  Caption       = "Site &Name:"
  DragMode      = 0
  Enabled       = -1
  ForeColor     = QBColor(0)
  Height        = Char(1)
  Left          = Char(1)
  MousePointer  = 0
  TabIndex      = 0
  Tag           = ""
  Top           = Char(1)
  Visible       = -1
  Width         = Char(11)

END
BEGIN Label Label11
  Alignment     = 0
  AutoSize      = 0
  BackColor     = QBColor(3)
  BorderStyle   = 0
  Caption       = "Site &Remarks:"
  DragMode      = 0
  Enabled       = -1
  ForeColor     = QBColor(0)
  Height        = Char(1)
  Left          = Char(1)
  MousePointer  = 0
  TabIndex      = 17
  Tag           = ""
  Top           = Char(13)
  Visible       = -1
  Width         = Char(14)

END
BEGIN Label Label4
  Alignment     = 0
  AutoSize      = 0
  BackColor     = QBColor(3)
  BorderStyle   = 0
  Caption       = "&Latitude"
  DragMode      = 0
  Enabled       = -1
  ForeColor     = QBColor(0)
  Height        = Char(1)
  Left          = Char(3)
  MousePointer  = 0
  TabIndex      = 9
  Tag           = ""
  Top           = Char(9)
  Visible       = -1
  Width         = Char(10)

END
BEGIN Label Label3
  Alignment     = 0

```

```

AutoSize      = 0
BackColor    = QBColor(3)
BorderStyle  = 0
Caption      = "Longitude"
DragMode     = 0
Enabled      = -1
ForeColor    = QBColor(0)
Height       = Char(1)
Left         = Char(38)
MousePointer = 0
TabIndex     = 13
Tag          = ""
Top          = Char(9)
Visible      = -1
Width        = Char(10)
END
OPTION EXPLICIT
'$INCLUDE: 'secpop90.bi'

'This form allows the user to view and edit site files.

SUB cmdClose_Click ()
    'This routine checks to see if the user has entered
    'valid input and then unloads the site form.  If the
    'user data is incorrect, the program generates
    'a warning.

    DIM temp AS INTEGER
    temp = verify_input(2)
    frmSite_Data.HIDE
END SUB

SUB cmdOpen_Click ()
    'This routine calls the function which opens a
    'previously created site file.

    CALL open_site
END SUB

SUB cmdPrint_Click ()
    'This routine calls a procedure which prints
    'the information in a site file

    CALL print_site
END SUB

SUB cmdSave_As_Click ()
    'This routine calls a procedure which saves a file
    'by a new name.

    CALL save_as_site

```

```

END SUB

SUB cmdSave_Click ()
    'This routine calls a procedure which saves a
    'site file.

    CALL save_site

END SUB

SUB Form_Unload (Cancel AS INTEGER)

    'This routine checks to see if the user has entered
    'valid input and then unloads the site form.  If the
    'user data is incorrect, the program generates
    'a warning.

    DIM temp AS INTEGER

    temp = verify_input(2)

    frmSite_Data.HIDE

END SUB

SUB txtLatitude_Degrees_KeyPress (keyascii AS INTEGER)

    'Verify that the latitude degrees are a number,
    'that the final number is two digits or shorter,
    'and mark the Site file as modified.

    IF ((keyascii < 48) OR (keyascii > 57) OR
    (LEN(frmSite_Data.txtLatitude_Degrees.text) >= 2)) AND (keyascii <> 8) AND keyascii
    <> 127 THEN

        keyascii = 0

    ELSE

        frmMain.mnuSave_SiteCom.Enabled = TRUE
        frmMain.mnuSave_As_SiteCom.Enabled = TRUE

        IF (INSTR(frmSite_Data.Caption, " (Modified)") = 0) THEN
            frmSite_Data.Caption = frmSite_Data.Caption + " (Modified)"
        END IF
        IF (INSTR(frmProblem_Data.txtSite_File_Name.text, " (Modified)") = 0) THEN
            frmProblem_Data.txtSite_File_Name.text = frmSite_Data.Caption
        END IF

    END IF

END SUB

SUB txtLatitude_Minutes_KeyPress (keyascii AS INTEGER)

    'For documentation, see latitude degrees.

    IF ((keyascii < 48) OR (keyascii > 57) OR
    (LEN(frmSite_Data.txtLatitude_Minutes.text) >= 2)) AND (keyascii <> 8) AND keyascii
    <> 127 THEN

        keyascii = 0

```

```

ELSE

    frmMain.mnuSave_SiteCom.Enabled = TRUE
    frmMain.mnuSave_As_SiteCom.Enabled = TRUE

    IF (INSTR(frmSite_Data.Caption, " (Modified)") = 0) THEN
        frmSite_Data.Caption = frmSite_Data.Caption + " (Modified)"
    END IF
    IF (INSTR(frmProblem_Data.txtSite_File_Name.text, " (Modified)") = 0) THEN
        frmProblem_Data.txtSite_File_Name.text = frmSite_Data.Caption
    END IF

END IF

END SUB

SUB txtLatitude_Seconds_KeyPress (keyascii AS INTEGER)

    'For documentation, see latitude degrees.

    IF ((keyascii < 48) OR (keyascii > 57) OR
(LEN(frmSite_Data.txtLatitude_Seconds.text) >= 2)) AND (keyascii <> 8) AND keyascii
<> 127 THEN

        keyascii = 0

    ELSE

        frmMain.mnuSave_SiteCom.Enabled = TRUE
        frmMain.mnuSave_As_SiteCom.Enabled = TRUE

        IF (INSTR(frmSite_Data.Caption, " (Modified)") = 0) THEN
            frmSite_Data.Caption = frmSite_Data.Caption + " (Modified)"
        END IF
        IF (INSTR(frmProblem_Data.txtSite_File_Name.text, " (Modified)") = 0) THEN
            frmProblem_Data.txtSite_File_Name.text = frmSite_Data.Caption
        END IF

    END IF

END SUB

SUB txtLongitude_Degrees_KeyPress (keyascii AS INTEGER)

    'Verify that the longitude degrees are a number,
    'that the final number is two digits or shorter,
    'and mark the Site file as modified.

    IF ((keyascii < 48) OR (keyascii > 57) OR
(LEN(frmSite_Data.txtLongitude_Degrees.text) >= 3)) AND (keyascii <> 8) AND keyascii
<> 127 THEN

        keyascii = 0

    ELSE

        frmMain.mnuSave_SiteCom.Enabled = TRUE
        frmMain.mnuSave_As_SiteCom.Enabled = TRUE

        IF (INSTR(frmSite_Data.Caption, " (Modified)") = 0) THEN

```

```

        frmSite_Data.Caption = frmSite_Data.Caption + " (Modified)"
    END IF
    IF (INSTR(frmProblem_Data.txtSite_File_Name.text, " (Modified)") = 0) THEN
        frmProblem_Data.txtSite_File_Name.text = frmSite_Data.Caption
    END IF

    END IF

END SUB

SUB txtLongitude_Minutes_KeyPress (keyascii AS INTEGER)
    'For documentation, see longitude degrees.

    IF ((keyascii < 48) OR (keyascii > 57) OR
(LEN(frmSite_Data.txtLongitude_Minutes.text) >= 2)) AND (keyascii <> 8) AND keyascii
<> 127 THEN

        keyascii = 0

    ELSE

        frmMain.mnuSave_SiteCom.Enabled = TRUE
        frmMain.mnuSave_As_SiteCom.Enabled = TRUE

        IF (INSTR(frmSite_Data.Caption, " (Modified)") = 0) THEN
            frmSite_Data.Caption = frmSite_Data.Caption + " (Modified)"
        END IF
        IF (INSTR(frmProblem_Data.txtSite_File_Name.text, " (Modified)") = 0) THEN
            frmProblem_Data.txtSite_File_Name.text = frmSite_Data.Caption
        END IF

    END IF

END SUB

SUB txtLongitude_Seconds_KeyPress (keyascii AS INTEGER)
    'For documentation, see longitude degrees.

    IF ((keyascii < 48) OR (keyascii > 57) OR
(LEN(frmSite_Data.txtLongitude_Seconds.text) >= 2)) AND (keyascii <> 8) AND keyascii
<> 127 THEN

        keyascii = 0

    ELSE

        frmMain.mnuSave_SiteCom.Enabled = TRUE
        frmMain.mnuSave_As_SiteCom.Enabled = TRUE

        IF (INSTR(frmSite_Data.Caption, " (Modified)") = 0) THEN
            frmSite_Data.Caption = frmSite_Data.Caption + " (Modified)"
        END IF
        IF (INSTR(frmProblem_Data.txtSite_File_Name.text, " (Modified)") = 0) THEN
            frmProblem_Data.txtSite_File_Name.text = frmSite_Data.Caption
        END IF

    END IF

END SUB

SUB txtSite_Name_KeyPress (keyascii AS INTEGER)

```

```

'Disable the return key in the site field, and mark
'the site form as modified.

IF keyascii = KEY_RETURN THEN

    keyascii = 0
    MSGBOX "Sorry, no hard returns allowed." + CHR$(13) + "Words will wrap
automatically.", MB_OK, "Error"

ELSE

    frmMain.mnuSave_SiteCom.Enabled = TRUE
    frmMain.mnuSave_As_SiteCom.Enabled = TRUE

    IF (INSTR(frmSite_Data.Caption, " (Modified)") = 0) THEN
        frmSite_Data.Caption = frmSite_Data.Caption + " (Modified)"
    END IF
    IF (INSTR(frmProblem_Data.txtSite_File_Name.text, " (Modified)") = 0) THEN
        frmProblem_Data.txtSite_File_Name.text = frmSite_Data.Caption
    END IF

END IF

END SUB

SUB txtSite_Remarks_KeyPress (keyascii AS INTEGER)

'Disable the enter key in the remarks field, and
'Mark the site form as modified.

IF keyascii = KEY_RETURN THEN

    keyascii = 0
    MSGBOX "Sorry, no hard returns allowed." + CHR$(13) + "Words will wrap
automatically.", MB_OK, "Error"

ELSE

    frmMain.mnuSave_SiteCom.Enabled = TRUE
    frmMain.mnuSave_As_SiteCom.Enabled = TRUE

    IF (INSTR(frmSite_Data.Caption, " (Modified)") = 0) THEN
        frmSite_Data.Caption = frmSite_Data.Caption + " (Modified)"
    END IF
    IF (INSTR(frmProblem_Data.txtSite_File_Name.text, " (Modified)") = 0) THEN
        frmProblem_Data.txtSite_File_Name.text = frmSite_Data.Caption
    END IF

END IF

END SUB

```

FILENAME: TABLE1.FRM

```
Version 1.00
BEGIN Form frmTable_1
    AutoRedraw      = -1
    BackColor       = QBColor(3)
    BorderStyle     = 1
    Caption         = "Population Table"
    ControlBox      = -1
    Enabled         = -1
    ForeColor       = QBColor(0)
    Height          = Char(21)
    Left            = Char(0)
    MaxButton       = 0
    MinButton       = 0
    MousePointer    = 0
    Tag             = ""
    Top             = Char(1)
    Visible         = -1
    Width           = Char(78)
    WindowState     = 0
    BEGIN Frame Frame1
        BackColor      = QBColor(3)
        Caption         = ""
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(18)
        Left            = Char(1)
        MousePointer    = 0
        TabIndex        = 31
        Tag             = ""
        Top             = Char(0)
        Visible         = -1
        Width           = Char(62)
    END
    BEGIN Label lblCsum_array
        Alignment       = 1
        AutoSize        = 0
        BackColor       = QBColor(3)
        BorderStyle     = 0
        Caption         = "123Sum789"
        DragMode        = 0
        Enabled         = -1
        ForeColor       = QBColor(0)
        Height          = Char(1)
        Index           = 4
        Left            = Char(51)
        MousePointer    = 0
        TabIndex        = 32
        Tag             = ""
        Top             = Char(12)
        Visible         = -1
        Width           = Char(9)
    END
    BEGIN Label lblCsum_array
        Alignment       = 1
        AutoSize        = 0
        BackColor       = QBColor(3)
        BorderStyle     = 0
        Caption         = "123Sum789"
        DragMode        = 0
```

```

        Enabled      = -1
        ForeColor   = QBColor(0)
        Height      = Char(1)
        Index       = 5
        Left        = Char(51)
        MousePointer = 0
        TabIndex    = 33
        Tag         = ""
        Top         = Char(14)
        Visible     = -1
        Width       = Char(9)
    END
    BEGIN Label lblCsum_array
        Alignment   = 1
        AutoSize    = 0
        BackColor   = QBColor(3)
        BorderStyle = 0
        Caption     = "123Sum789"
        DragMode    = 0
        Enabled     = -1
        ForeColor   = QBColor(0)
        Height      = Char(1)
        Index       = 3
        Left        = Char(51)
        MousePointer = 0
        TabIndex    = 24
        Tag         = ""
        Top         = Char(10)
        Visible     = -1
        Width       = Char(9)
    END
    BEGIN Label lblCsum_array
        Alignment   = 1
        AutoSize    = 0
        BackColor   = QBColor(3)
        BorderStyle = 0
        Caption     = "123Sum789"
        DragMode    = 0
        Enabled     = -1
        ForeColor   = QBColor(0)
        Height      = Char(1)
        Index       = 2
        Left        = Char(51)
        MousePointer = 0
        TabIndex    = 25
        Tag         = ""
        Top         = Char(8)
        Visible     = -1
        Width       = Char(9)
    END
    BEGIN Label lblCsum_array
        Alignment   = 1
        AutoSize    = 0
        BackColor   = QBColor(3)
        BorderStyle = 0
        Caption     = "123Sum789"
        DragMode    = 0
        Enabled     = -1
        ForeColor   = QBColor(0)
        Height      = Char(1)
        Index       = 1
        Left        = Char(51)
        MousePointer = 0
    END

```

```

TabIndex      = 26
Tag          = ""
Top          = Char(6)
Visible      = -1
Width        = Char(9)
END
BEGIN Label lblCsum_array
Alignment    = 1
AutoSize     = 0
BackColor   = QBColor(3)
BorderStyle = 0
Caption     = "123Sum789"
DragMode     = 0
Enabled      = -1
ForeColor   = QBColor(0)
Height       = Char(1)
Index        = 0
Left         = Char(51)
MousePointer = 0
TabIndex     = 27
Tag          = ""
Top          = Char(4)
Visible      = -1
Width        = Char(9)
END
BEGIN Label lblTotal
Alignment    = 1
AutoSize     = 0
BackColor   = QBColor(3)
BorderStyle = 0
Caption     = "SumSum789"
DragMode     = 0
Enabled      = -1
ForeColor   = QBColor(0)
Height       = Char(1)
Left         = Char(51)
MousePointer = 0
TabIndex     = 34
Tag          = ""
Top          = Char(16)
Visible      = -1
Width        = Char(9)
END
BEGIN Label Label21
Alignment    = 0
AutoSize     = -1
BackColor   = QBColor(3)
BorderStyle = 0
Caption     = "Sum"
DragMode     = 0
Enabled      = -1
ForeColor   = QBColor(0)
Height       = Char(1)
Left         = Char(54)
MousePointer = 0
TabIndex     = 9
Tag          = ""
Top          = Char(2)
Visible      = -1
Width        = Char(3)
END
BEGIN Label lblCumulative_Header
Alignment    = 0

```

```

AutoSize      = -1
BackColor    = QBColor(3)
BorderStyle  = 0
Caption      = "Sum"
DragMode     = 0
Enabled      = -1
ForeColor    = QBColor(0)
Height       = Char(1)
Left         = Char(3)
MousePointer = 0
TabIndex     = 14
Tag          = ""
Top          = Char(16)
Visible      = -1
Width        = Char(3)

```

END

```

BEGIN Label lblSector_array
  Alignment    = 0
  AutoSize    = 0
  BackColor   = QBColor(3)
  BorderStyle = 0
  Caption     = "A"
  DragMode    = 0
  Enabled     = -1
  ForeColor   = QBColor(0)
  Height      = Char(1)
  Index       = 0
  Left         = Char(3)
  MousePointer = 0
 TabIndex     = 15
  Tag          = ""
  Top          = Char(4)
  Visible     = -1
  Width        = Char(3)

```

END

```

BEGIN Label lblSector_array
  Alignment    = 0
  AutoSize    = 0
  BackColor   = QBColor(3)
  BorderStyle = 0
  Caption     = "B"
  DragMode    = 0
  Enabled     = -1
  ForeColor   = QBColor(0)
  Height      = Char(1)
  Index       = 1
  Left         = Char(3)
  MousePointer = 0
 TabIndex     = 16
  Tag          = ""
  Top          = Char(6)
  Visible     = -1
  Width        = Char(3)

```

END

```

BEGIN Label lblSector_array
  Alignment    = 0
  AutoSize    = 0
  BackColor   = QBColor(3)
  BorderStyle = 0
  Caption     = "C"
  DragMode    = 0
  Enabled     = -1
  ForeColor   = QBColor(0)

```

```

Height      = Char(1)
Index       = 2
Left        = Char(3)
MousePointer = 0
TabIndex    = 17
Tag         = ""
Top         = Char(8)
Visible     = -1
Width       = Char(3)

END
BEGIN Label lblSector_array
Alignment   = 0
AutoSize    = 0
BackColor   = QBColor(3)
BorderStyle = 0
Caption     = "D"
DragMode    = 0
Enabled     = -1
ForeColor   = QBColor(0)
Height      = Char(1)
Index       = 3
Left        = Char(3)
MousePointer = 0
TabIndex    = 18
Tag         = ""
Top         = Char(10)
Visible     = -1
Width       = Char(3)

END
BEGIN Label lblSector_array
Alignment   = 0
AutoSize    = 0
BackColor   = QBColor(3)
BorderStyle = 0
Caption     = "E"
DragMode    = 0
Enabled     = -1
ForeColor   = QBColor(0)
Height      = Char(1)
Index       = 4
Left        = Char(3)
MousePointer = 0
TabIndex    = 19
Tag         = ""
Top         = Char(12)
Visible     = -1
Width       = Char(3)

END
BEGIN Label lblSector_array
Alignment   = 0
AutoSize    = 0
BackColor   = QBColor(3)
BorderStyle = 0
Caption     = "F"
DragMode    = 0
Enabled     = -1
ForeColor   = QBColor(0)
Height      = Char(1)
Index       = 5
Left        = Char(3)
MousePointer = 0
TabIndex    = 28
Tag         = ""

```

```

        Top          = Char(14)
        Visible      = -1
        Width        = Char(3)
END
BEGIN Label lblRsum_array
        Alignment    = 1
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 0
        Caption      = "Sum456789"
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(1)
        Index        = 0
        Left         = Char(9)
        MousePointer = 0
        TabIndex     = 20
        Tag          = ""
        Top          = Char(16)
        Visible      = -1
        Width        = Char(9)
END
BEGIN Label lblRsum_array
        Alignment    = 1
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 0
        Caption      = "Sum456789"
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(1)
        Index        = 1
        Left         = Char(19)
        MousePointer = 0
        TabIndex     = 21
        Tag          = ""
        Top          = Char(16)
        Visible      = -1
        Width        = Char(9)
END
BEGIN Label lblRsum_array
        Alignment    = 1
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 0
        Caption      = "Sum456789"
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(1)
        Index        = 2
        Left         = Char(29)
        MousePointer = 0
        TabIndex     = 22
        Tag          = ""
        Top          = Char(16)
        Visible      = -1
        Width        = Char(9)
END
BEGIN Label lblRsum_array
        Alignment    = 1

```

```

AutoSize      = 0
BackColor    = QBColor(3)
BorderStyle  = 0
Caption      = "Sum456789"
DragMode     = 0
Enabled      = -1
ForeColor    = QBColor(0)
Height       = Char(1)
Index        = 3
Left         = Char(39)
MousePointer = 0
TabIndex     = 23
Tag          = ""
Top          = Char(16)
Visible      = -1
Width        = Char(9)
END
BEGIN HScrollBar hsbRadii
Attached      = 0
DragMode     = 0
Enabled      = -1
Height       = Char(1)
LargeChange   = 1
Left         = Char(2)
Max          = 50
Min          = 1
MousePointer = 0
SmallChange   = 1
TabIndex     = 29
TabStop      = -1
Tag          = ""
Top          = Char(17)
Value        = 1
Visible      = -1
Width        = Char(60)
END
BEGIN VScrollBar vsbSectors
Attached      = 0
DragMode     = 0
Enabled      = -1
Height       = Char(16)
LargeChange   = 1
Left         = Char(62)
Max          = 16
Min          = 1
MousePointer = 0
SmallChange   = 1
TabIndex     = 30
TabStop      = -1
Tag          = ""
Top          = Char(1)
Value        = 1
Visible      = -1
Width        = Char(1)
END
BEGIN Label lblDistance_array
Alignment    = 1
AutoSize     = 0
BackColor    = QBColor(3)
BorderStyle  = 0
Caption      = "1000.0000"
DragMode     = 0
Enabled      = -1

```

```

ForeColor      = QBColor(0)
Height        = Char(1)
Index         = 0
Left          = Char(9)
MousePointer  = 0
TabIndex      = 10
Tag           = ""
Top           = Char(2)
Visible       = -1
Width         = Char(9)
END
BEGIN Label lblDistance_array
Alignment      = 1
AutoSize       = 0
BackColor     = QBColor(3)
BorderStyle   = 0
Caption       = "2000.0000"
DragMode      = 0
Enabled        = -1
ForeColor     = QBColor(0)
Height        = Char(1)
Index         = 1
Left          = Char(19)
MousePointer  = 0
TabIndex      = 11
Tag           = ""
Top           = Char(2)
Visible       = -1
Width         = Char(9)
END
BEGIN Label lblDistance_array
Alignment      = 1
AutoSize       = 0
BackColor     = QBColor(3)
BorderStyle   = 0
Caption       = "3000.0000"
DragMode      = 0
Enabled        = -1
ForeColor     = QBColor(0)
Height        = Char(1)
Index         = 2
Left          = Char(29)
MousePointer  = 0
TabIndex      = 12
Tag           = ""
Top           = Char(2)
Visible       = -1
Width         = Char(9)
END
BEGIN Label lblDistance_array
Alignment      = 1
AutoSize       = 0
BackColor     = QBColor(3)
BorderStyle   = 0
Caption       = "4000.0000"
DragMode      = 0
Enabled        = -1
ForeColor     = QBColor(0)
Height        = Char(1)
Index         = 3
Left          = Char(39)
MousePointer  = 0
TabIndex      = 13

```

```

        Tag          = ""
        Top          = Char(2)
        Visible     = -1
        Width        = Char(9)
    END
    BEGIN Label lblValue_array
        Alignment    = 0
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 1
        Caption      = " 123456789 123456789 123456789 123456789 "
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 0
        Left         = Char(7)
        MousePointer = 0
        TabIndex     = 2
        Tag          = ""
        Top          = Char(3)
        Visible     = -1
        Width        = Char(43)
    END
    BEGIN Label lblValue_array
        Alignment    = 0
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 1
        Caption      = " 123456789 123456789 123456789 123456789 "
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 1
        Left         = Char(7)
        MousePointer = 0
        TabIndex     = 3
        Tag          = ""
        Top          = Char(5)
        Visible     = -1
        Width        = Char(43)
    END
    BEGIN Label lblValue_array
        Alignment    = 0
        AutoSize     = 0
        BackColor    = QBColor(3)
        BorderStyle  = 1
        Caption      = " 123456789 123456789 123456789 123456789 "
        DragMode     = 0
        Enabled      = -1
        ForeColor    = QBColor(0)
        Height       = Char(3)
        Index        = 2
        Left         = Char(7)
        MousePointer = 0
        TabIndex     = 4
        Tag          = ""
        Top          = Char(7)
        Visible     = -1
        Width        = Char(43)
    END
    BEGIN Label lblValue_array

```

```

        Alignment      = 0
        AutoSize       = 0
        BackColor      = QBColor(3)
        BorderStyle    = 1
        Caption        = " 123456789 123456789 123456789 123456789 "
        DragMode       = 0
        Enabled        = -1
        ForeColor      = QBColor(0)
        Height         = Char(3)
        Index          = 3
        Left           = Char(7)
        MousePointer   = 0
        TabIndex       = 5
        Tag            = ""
        Top            = Char(9)
        Visible        = -1
        Width          = Char(43)
    END
    BEGIN Label lblValue_array
        Alignment      = 0
        AutoSize       = 0
        BackColor      = QBColor(3)
        BorderStyle    = 1
        Caption        = " 123456789 123456789 123456789 123456789 "
        DragMode       = 0
        Enabled        = -1
        ForeColor      = QBColor(0)
        Height         = Char(3)
        Index          = 4
        Left           = Char(7)
        MousePointer   = 0
        TabIndex       = 6
        Tag            = ""
        Top            = Char(11)
        Visible        = -1
        Width          = Char(43)
    END
    BEGIN Label lblValue_array
        Alignment      = 0
        AutoSize       = 0
        BackColor      = QBColor(3)
        BorderStyle    = 1
        Caption        = " 123456789 123456789 123456789 123456789 "
        DragMode       = 0
        Enabled        = -1
        ForeColor      = QBColor(0)
        Height         = Char(3)
        Index          = 5
        Left           = Char(7)
        MousePointer   = 0
        TabIndex       = 7
        Tag            = ""
        Top            = Char(13)
        Visible        = -1
        Width          = Char(43)
    END
    BEGIN CommandButton cmdexit
        BackColor      = QBColor(3)
        Cancel         = 0
        Caption        = "&Close"
        Default        = -1
        DragMode       = 0
        Enabled        = -1

```

```

        Height      = Char(3)
        Left       = Char(65)
        MousePointer = 0
        TabIndex    = 1
        TabStop     = -1
        Tag         = ""
        Top         = Char(6)
        Visible    = -1
        Width       = Char(9)
    END
    BEGIN CommandButton cmdprint
        BackColor   = QBColor(3)
        Cancel      = 0
        Caption     = "&Print"
        Default     = 0
        DragMode    = 0
        Enabled     = -1
        Height      = Char(3)
        Left        = Char(65)
        MousePointer = 0
        TabIndex    = 0
        TabStop     = -1
        Tag         = ""
        Top         = Char(9)
        Visible    = -1
        Width       = Char(9)
    END
    BEGIN Label lblDistance_Heading
        Alignment   = 2
        AutoSize    = 0
        BackColor   = QBColor(3)
        BorderStyle = 0
        Caption     = "Radii (Kilometers)"
        DragMode    = 0
        Enabled     = -1
        ForeColor   = QBColor(0)
        Height      = Char(1)
        Left        = Char(20)
        MousePointer = 0
        TabIndex    = 8
        Tag         = ""
        Top         = Char(1)
        Visible    = -1
        Width       = Char(18)
    END
END
OPTION EXPLICIT

'$INCLUDE: 'secpop90.bi'

'This is the actual table form for table 1.  See rptable_1 for more
'info.

SUB cmdExit_Click ()
    'Remove table 1 from memory.

    UNLOAD frmTable_1

END SUB

sub cmdPrint_Click ()

```

```

'This routine prints the displayed population table.

dim Forecolor as integer, Backcolor as integer
dim Copies as integer, Cancel as integer
dim quote as string, comma as string
dim blank as string, degree as string
dim pages as integer, radius as integer
dim first_radius as integer, last_radius as integer
dim i as integer, j as integer, k as integer
dim l as integer, m as integer, n as integer
dim sum as long

'Set quote and comma strings to null if MACCS input file format is
'selected or to "" and ',' if comma separated variable (CSV) format
'is selected.

if (frmSetup.optMACCS.Value) then
    quote = ""
    comma = ""
    blank = " "
elseif (frmSetup.optCSV.Value) then
    quote = """
    comma = ","
    blank = ""
else
    quote = ""
    comma = ""
    blank = " "
end if

'Set degree character.

degree = chr$(248)

'Set colors for Dialog box

BackColor = WHITE
ForeColor = BLACK

'Call the standard print dialog.

call FilePrint(Copies, ForeColor, BackColor, Cancel)

'If cancel is not selected by user in the printer dialogue box,
'then proceed.

if (not Cancel) then

    'Enable local error checking.

    on local error goto table1_printer_error

    'If the user is printing to a printer adjust the margins so that
    'there are no extra line feeds after the lines that have 80
    'columns otherwise prepend the output path to the printer
    'target.

    if((printer.PrintTarget = "LPT1:") or -
       (printer.PrintTarget = "LPT2:") or -
       (printer.PrintTarget = "LPT3:")) then

        width printer.PrintTarget, 81

```

```

else

    printer.PrintTarget = frmSetup.txtOutput_path.text + _
        "\\" + printer.PrintTarget

end if

'Determine the number of "pages" that need to be printed per
'copy. A single page is considered to be 16 directions and a
'sum row long, by at most 6 radii or 5 radii and a sum column.

pages = (number_of_radii + 1) \ 6
if((number_of_radii + 1) mod 6) <> 0) then
    pages = pages + 1
end if

'Execute loop once for each copy requested by the user.

for i = 1 to Copies step 1

    'Print the first line.

    printer.print quote;
    printer.print "SECPOP90 V2.3 ";
    if (table_type = 1) then
        printer.print "Population Data";
    else
        printer.print "Cumulative Population Data";
    end if
    printer.print quote;
    printer.print

    'Print the second line.

    printer.print quote;
    printer.print "Date:";
    printer.print quote; comma; blank;
    printer.print quote;
    printer.print format$(now, "mm/dd/yyyy");
    printer.print quote; comma; blank; blank; blank;
    printer.print quote;
    printer.print "Time:";
    printer.print quote; comma; blank;
    printer.print quote;
    printer.print format$(now, "hh:mm:ss");
    printer.print quote;
    printer.print

    'Print the third line.

    printer.print quote;
    printer.print "Site Name:";
    printer.print quote; comma; blank;
    printer.print quote;
    printer.print
        ltrim$(rtrim$(frmSite_data.txtSite_name.Text));
    printer.print quote;
    printer.print

    'Print the fourth line.

    printer.print quote;

```

```

printer.print "Latitude:";
printer.print quote; comma; blank;
printer.print using "##";
    val(frmSite_data.txtLatitude_degrees.Text);
printer.print comma;
printer.print quote;
printer.print degree;
printer.print quote; comma;
printer.print using "##";
    val(frmSite_data.txtLatitude_minutes.Text);
printer.print comma;
printer.print quote;
printer.print "'";
printer.print quote; comma;
printer.print using "##";
    val(frmSite_data.txtLatitude_seconds.Text);
printer.print comma;
printer.print quote;
printer.print "'";
printer.print quote; comma; blank; blank; blank;
printer.print quote;
printer.print "Longitude:";
printer.print quote; comma; blank;
printer.print using "###";
    val(frmSite_data.txtLongitude_degrees.Text);
printer.print comma;
printer.print quote;
printer.print degree;
printer.print quote; comma;
printer.print using "##";
    val(frmSite_data.txtLongitude_minutes.Text);
printer.print comma;
printer.print quote;
printer.print "'";
printer.print quote; comma;
printer.print using "##";
    val(frmSite_data.txtLongitude_seconds.Text);
printer.print comma;
printer.print quote;
printer.print "'";
printer.print quote;
printer.print

'Print the fifth line.

printer.print quote;
printer.print "Population Multiplier:";
printer.print quote; comma; blank;
printer.print using "#####.#####";
    val(frmProblem_data.txtPopulation_multiplier.Text);
printer.print

'Print out the "pages" of population data.

'Initialize the radius index to point to the first radius.

radius = 1

for j = 1 to pages step 1

    'Print out the radii header row.
    printer.print
    printer.print

```

```

printer.print quote;
printer.print "Radius";
if (frmProblem_Data.optKilometers.value) then
    printer.print "km)";
else
    printer.print "mi)";
end if
printer.print quote; comma;

'Print out the radii, setting the loop min and max to
'the radii to be printed for this page. Note: the final
'radii column is the sum column.

first_radius = radius
if((radius + 5) <= (number_of_radii + 1)) then
    last_radius = radius + 5
else
    last_radius = number_of_radii + 1
end if

for k = first_radius to last_radius step 1
    if (k <> (number_of_radii + 1)) then
        printer.print using "#####.###"; _
            radial_distance(k);
        if (k <> last_radius) then
            printer.print comma;
        end if
    else
        printer.print "      "; blank;
        printer.print quote;
        printer.print "Sum";
        printer.print quote;
    end if
next k

printer.print
printer.print

'Print out the direction and the (cumulative)
'population values. The last column is the sum of the
'population values.

for k = 1 to number_of_segments step 1
    printer.print quote;
    printer.print using "\      \"; directions(k);
    printer.print quote; comma;
    for l = first_radius to last_radius step 1
        if (l <> (number_of_radii + 1)) then
            if (table_type = 1) then
                printer.print using "#####"; _
                    sector_population(k, l);
            else
                sum = 0
                for m = 1 to l step 1
                    sum = sum + sector_population(k, m)
                next m
                printer.print using "#####"; sum;
            end if
            if (l <> last_radius) then
                printer.print comma;
            end if
        else
            sum = 0

```

```

        for m = 1 to number_of_radii step 1
            sum = sum + sector_population(k, m)
        next m
        printer.print using "#####"; sum;
    end if
next l
printer.print
next k

printer.print quote;
printer.print "Sum      ";
printer.print quote; comma;
for l = first_radius to last_radius step 1
    if (l <> number_of_radii + 1) then
        sum = 0
        if (table_type = 1) then
            for m = 1 to number_of_segments step 1
                sum = sum + sector_population(m, l)
            next m
        else
            for m = 1 to number_of_segments step 1
                for n = 1 to l step 1
                    sum = sum + sector_population(m, n)
                next n
            next m
        end if
        printer.print using "#####"; sum;
        if (l <> last_radius) then
            printer.print comma;
        end if
    else
        sum = 0
        for m = 1 to number_of_segments step 1
            for n = 1 to number_of_radii step 1
                sum = sum + sector_population(m, n)
            next n
        next m
        printer.print using "#####"; sum;
    end if
next l
printer.print

radius = last_radius + 1

next j

printer.newpage

next i

printer.enddoc

end if

exit sub

table1_printer_error:
    'If there is a problem, tell the user and exit the subroutine.

msgbox error$(err)

exit sub

```

```
end sub

SUB Form_Load ()
    'Perform first call to paint table to initialize the table.

    UNLOAD frmDisclaimer

    x_position = 1
    y_position = 1
    CALL paint_table

END SUB

SUB hsbRadius_Change ()
    x_position = hsbRadius.value
    CALL paint_table

END SUB

SUB vsbSectors_Change ()
    y_position = vsbSectors.value
    CALL paint_table

END SUB
```

**FILENAME: UTILTIY.BAS**

OPTION EXPLICIT

'\$INCLUDE: 'secpop90.bi'

'Various general purpose utilities:

'1. Print stuff to a non-Form screen.  
'2. Print stuff to the printer.  
'3. Verify the user's input in one of several forms.  
'4. Copy from one file to another.

SUB parse\_path\_and\_file (caption AS STRING, path\_name AS STRING, file\_name AS STRING)

'This routine removes the path from a file name. This is necessary when  
'passing a default filename to a file dialog box.

DIM temp AS STRING  
DIM modified\_location AS INTEGER  
DIM slash\_found AS INTEGER, slash\_location AS INTEGER

'Remove the modified specifier from the file name if necessary.

modified\_location = INSTR(caption, " (Modified)")  
IF modified\_location <> 0 THEN  
 temp = MID\$(caption, 1, modified\_location - 1)  
ELSE  
 temp = caption  
END IF

'Search backward from the end of the string to the first slash,  
'and record the position of the first slash.

slash\_found = FALSE  
slash\_location = LEN(temp)  
WHILE ((NOT slash\_found) AND (slash\_location > 0))  
 IF MID\$(temp, slash\_location, 1) = "\" THEN  
 slash\_found = TRUE  
 ELSE  
 slash\_location = slash\_location - 1  
 END IF

WEND

'If a slash was found, break the name into two parts - path and file.

IF slash\_found = TRUE THEN  
 path\_name = MID\$(temp, 1, slash\_location - 1)  
 file\_name = MID\$(temp, slash\_location + 1)

'Otherwise, return the unchanged filename.

ELSE  
 path\_name = ""  
 file\_name = temp  
END IF

END SUB

SUB print\_text (text AS STRING, text\_width AS INTEGER, left\_margin AS INTEGER)

'This routine prints formatted text strings in a non-form environment.

```

DIM next_line AS STRING, character AS STRING
DIM character_location AS INTEGER, character_found AS INTEGER

'Execute loop while the lenght of text is greater
'than the user defined size.

WHILE (LEN(text) > text_width)

    'Put next string of length user has requested into next line.

    next_line = LEFT$(text, text_width)
    character_found = FALSE
    character_location = text_width

    'Scan input string for special characters ( <cr>, -, and " ").

    WHILE ((NOT character_found) AND (character_location > 0))
        character = MID$(next_line, character_location, 1)
        IF ((character = " ") OR (character = "-") OR (character = CHR$(13)))
THEN
            character_found = TRUE
        ELSE
            character_location = character_location - 1
        END IF
    WEND

    'Process special characters found in string.

    IF character_found THEN
        PRINTER.PRTTAB(left_margin); LEFT$(next_line, character_location -
1);
        IF character = "-" THEN
            PRINTER.PRINT character
        ELSE
            PRINTER.PRINT
        END IF
        character_found = FALSE
        text = MID$(text, character_location + 1)
    ELSE
        'If no characters found, then print string.

        PRINTER.PRINT TAB(left_margin); next_line
        text = MID$(text, text_width + 1)

    END IF

    WEND

    PRINTER.PRINT TAB(left_margin); text

END SUB

FUNCTION verify_input (which_form AS INTEGER) AS INTEGER

    DIM filenum AS INTEGER
    DIM tempstring AS STRING, i AS INTEGER, j AS INTEGER

    'Define constants representing which form is calling for input
    'verification.

```

```

CONST PROBLEM = 1, SITE = 2, REGIONAL_CALC = 3

'Decide which form is calling, analyze the input on that form,
'display a message if necessary, and return a value to the calling
'function.

verify_input = 0

SELECT CASE which_form

CASE PROBLEM

    'Enable local error checking for the new problem form to deal
    'with disk related errors check for the file then reset local
    'error checking.

    ON LOCAL ERROR GOTO problem_bad_file_name
    filenum = FREEFILE
    OPEN frmProblem_Data.txtSite_File_Name.text FOR INPUT AS #filenum
    CLOSE filenum
    ON LOCAL ERROR GOTO 0

    IF VAL(frmProblem_Data.txtPopulation_multiplier.text) <= 0 THEN
        MSGBOX "Error: Population multiplier must greater than 0.  "
        verify_input = -1
    ELSEIF VAL(frmProblem_Data.txtPopulation_multiplier.text) > 9999.9999
THEN
        MSGBOX "Error: Population multiplier must be less than 10000"
        verify_input = -1
    ELSEIF (radial_distance(1) <= 0) THEN
        MSGBOX "Error: Radial distance 1 must be greater than 0."
        verify_input = -2
    ELSEIF (radial_distance(2) <= 0) THEN
        MSGBOX "Error: Radial distance 2 must be greater than 0."
        verify_input = -3
    ELSEIF (VAL(frmProblem_Data.lblNumber_of_regions.caption) < 2 OR
VAL(frmProblem_Data.lblNumber_of_regions.caption) > 99) THEN
        MSGBOX "Invalid Economic region settings"
        verify_input = -4
    ELSE
        'Everything's O.K.
        verify_input = 1
    END IF

CASE SITE

    IF (VAL(frmSite_Data.txtLatitude_Degrees.text) > 47) OR
(VAL(frmSite_Data.txtLatitude_Degrees.text) < 24) OR
(VAL(frmSite_Data.txtLongitude_Degrees.text) > 172) OR
(VAL(frmSite_Data.txtLongitude_Degrees.text) < 67) THEN

```

```

MSGBOX "Warning: The Site that you have specified is outside of the
continental United States. Data near the site may not be available."
verify_input = -1

ELSE

'Everything's O.K.
verify_input = 1

END IF

case regional_calc

if ((val(frmMake_a_circle.txtRadius.text) < 1) or
(val(frmMake_a_circle.txtRadius.text) > 170)) then

msgbox _
"Radius of Circles value must be greater than or" +
chr$(13) + chr$(10) +
"equal to 1 mile and less than or equal to 170" +
chr$(13) + chr$(10) +
"miles.", MB_OK, "Error"

verify_input = 0

elseif ((val(frmMake_a_circle.txtSpacing.text) < 1) or
(val(frmMake_a_circle.txtSpacing.text) > 340)) then

msgbox _
"Longitudinal Spacing of Circles value must" +
chr$(13) + chr$(10) +
"be greater than or equal to 1 mile and less" +
chr$(13) + chr$(10) +
"than or equal to 340 miles.", MB_OK, "Error"

verify_input = -1

elseif ((val(frmMake_a_circle.txtThreshold.text) < 1) or
(val(frmMake_a_circle.txtThreshold.text) > 20000)) then

msgbox _
"Population Density Threshold value must be" +
chr$(13) + chr$(10) +
"greater than or equal to 1 person per square" +
chr$(13) + chr$(10) +
"mile and less than or equal to 20000 persons" +
chr$(13) + chr$(10) +
"per square mile.", MB_OK, "Error"

verify_input = -2

elseif (frmMake_a_circle.txtInput_map.text = "") then

msgbox _
"Input Map Name missing. Please enter a valid" +
chr$(13) + chr$(10) +
"filename.", MB_OK, "Error"

verify_input = -3

else

'Enable local error checking for regional calc form

```

```

        on local error goto regional_file_error
        filenum = freefile
        open frmMake_a_circle.txtinput_map.text _
            for input as filenum
        close filenum
        on local error goto 0
        'If we got here then everything's O.K.
        verify_input = 1

    end if

END SELECT

EXIT FUNCTION

problem_bad_file_name:

MSGBOX "Error: Unable to open site file - check file name. "
verify_input = 0
EXIT FUNCTION

regional_file_error:

msgbox _
    "Input Map Name invalid. Please enter a valid" + _
    chr$(13) + chr$(10) +
    "filename.", MB_OK, "Error"

verify_input = -3

exit function

END FUNCTION

sub copy_file(input_filename as string, output_filename as string)

    dim i as long
    dim my_byte as string * 1
    dim input_filenumber as integer, output_filenumber as integer

    on local error goto error_return

    input_filenumber = freefile
    open input_filename for binary access read as #input_filenumber

    output_filenumber = freefile

    'Truncate output file to 0 length if it already existed.
    open output_filename for output access write as #output_filenumber
    close(output_filenumber)

    open output_filename for binary access write as #output_filenumber

    for i = 1 to lof(input_filenumber) step 1
        get #input_filenumber, , my_byte
        put #output_filenumber, , my_byte
    next i

    close(input_filenumber)
    close(output_filenumber)

normal_return:

```

```
exit sub

error_return:
    msgbox error$(err)
    exit sub

end sub
```

**FILENAME: SECPOP90.BI**

```
'$INCLUDE: 'types.bi'  
'$INCLUDE: 'declare.bi'  
'$INCLUDE: 'const.bi'  
'$INCLUDE: 'common.bi'  
'$INCLUDE: 'forms.bi'
```

**FILENAME: TYPES.BI**

```
TYPE economic_data  
  
    region_area AS LONG  
    region_frmfrc AS SINGLE  
    region_dpf AS SINGLE  
    region_asfp AS SINGLE  
    region_vfrm AS SINGLE  
    region_vnfrm AS SINGLE  
  
END TYPE
```

**FILENAME: FORMS.BI**

```
'$FORM frmCalculate  
'$FORM frmCmnDlg  
'$FORM frmDensity  
'$FORM frmDisclaimer  
'$FORM frmMain  
'$FORM frmMake_a_Circle  
'$FORM frmOutput  
'$FORM frmProblem_Data  
'$FORM frmRegion  
'$FORM frmSetup  
'$FORM frmSite_data  
'$FORM frmTable_1
```

**FILENAME: DECLARE.BI**

```
DECLARE FUNCTION DIST (X, Y)
DECLARE FUNCTION verify_input (which_form AS INTEGER) AS INTEGER
DECLARE SUB browse_site_file ()
DECLARE SUB cmdSort_Click ()
DECLARE SUB copy_file(input_filename as string, _
                      output_filename as string)
DECLARE SUB default_regions ()
DECLARE SUB FileOpen (filename AS STRING, pathname AS STRING,
                      DefaultExt AS STRING, DialogTitle AS STRING, forecolor AS INTEGER, _
                      backcolor AS INTEGER, Flags AS INTEGER, Cancel AS INTEGER)
DECLARE SUB FilePrint (Copies AS INTEGER, ForeColor AS INTEGER, _
                      BackColor AS INTEGER, Cancel AS INTEGER)
DECLARE SUB FileSave (FileName AS STRING, PathName AS STRING,
                      DefaultExt AS STRING, DialogTitle AS STRING, ForeColor AS INTEGER, _
                      BackColor AS INTEGER, Flags AS INTEGER, Cancel AS INTEGER)
DECLARE SUB filOpenList_DblClick ()
DECLARE SUB GETDIR (Y, DIR AS INTEGER)
DECLARE SUB GETDIS (Y AS SINGLE)
DECLARE SUB GETRAD (Y, NUMRAD AS INTEGER, RAD AS INTEGER)
DECLARE SUB new_problem ()
DECLARE SUB new_site ()
DECLARE SUB open_problem ()
DECLARE SUB open_site ()
DECLARE SUB open_site_from_problem (file_name AS STRING)
DECLARE SUB paint_region (x_offset AS INTEGER, y_offset AS INTEGER)
DECLARE SUB paint_table ()
DECLARE SUB paint_region (x AS INTEGER, y AS INTEGER)
DECLARE SUB parse_path_and_file (Caption AS STRING, _
                                 path_name AS STRING, file_name AS STRING)
DECLARE SUB pointr (BYVAL BW AS SINGLE, census_file AS LONG, Y,
                   POP AS INTEGER, area AS LONG, county_code AS INTEGER, i AS LONG)
DECLARE SUB popcalc ()
DECLARE SUB popdensity (exceeded AS INTEGER)
DECLARE SUB print_graphic (text AS STRING, text_width AS INTEGER, _
                           centered AS INTEGER)
DECLARE SUB print_MACCS_input_file ()
DECLARE SUB print_problem ()
DECLARE SUB print_site ()
DECLARE SUB print_text (text AS STRING, text_width AS INTEGER, _
                       left_margin AS INTEGER)
DECLARE SUB rosette ()
DECLARE SUB save_as_problem ()
DECLARE SUB save_as_site ()
DECLARE SUB save_problem ()
DECLARE SUB save_site ()
DECLARE SUB txtRadial_Distance1_LostFocus ()
DECLARE SUB txtRadial_Distance3_LostFocus ()
```

**FILENAME: CONST.BI**

```
'Microsoft Visual Basic constants. Unused (by SECPOP90) constants are
'commented out to save memory.

' -----
' Visual Basic for MS-DOS Constant Include File

' Include file that contains constant definitions
' for enumerated form or control property values and
' event procedure, method, and function parameter values.

' This file can be included and used as is in your applications.
' Each constant definition reduces the amount of memory available
' for your application, however, so for best results, include only
' those constant definitions you plan to use in your application.
' In addition, some of the constant definitions below may conflict with
' variable definitions in your existing programs.

' Some constants below are commented out because they
' have duplicates (for example, NONE appears in several
' places).

' Copyright (C) 1982-1992 Microsoft Corporation

' You have a royalty-free right to use, modify, reproduce
' and distribute the sample applications and toolkits provided with
' Visual Basic for MS-DOS (and/or any modified version)
' in any way you find useful, provided that you agree that
' Microsoft has no warranty, obligations or liability for
' any of the sample applications or toolkits.
-----
```

```
' -----
' General
' -----

' Booleans
CONST TRUE  = -1
CONST FALSE = 0

' -----
' Event parameters
' -----

' Button and Shift (KeyDown, KeyUp, MouseDown, MouseMove, MouseUp)
'CONST SHIFT_MASK = 1
'CONST CTRL_MASK = 2
'CONST ALT_MASK = 4
'CONST LEFT_BUTTON = 1
'CONST RIGHT_BUTTON = 2

' KeyCode (KeyDown, KeyUp)
'CONST KEY_BACK = 8
'CONST KEY_TAB = 9
'CONST KEY_CLEAR = 12
CONST KEY_RETURN = 13           ' Enter key
'CONST KEY_SHIFT = 16
'CONST KEY_CONTROL = 17
'CONST KEY_MENU = 18           ' Alt key
'CONST KEY_PAUSE = 19
```

```

'CONST KEY_CAPITAL = 20          ' Caps lock key
'CONST KEY_ESCAPE = 27
CONST KEY_SPACE = 32
'CONST KEY_PRIOR = 33          ' Page up key
'CONST KEY_NEXT = 34           ' Page down key
'CONST KEY_END = 35
'CONST KEY_HOME = 36
'CONST KEY_LEFT = 37
CONST KEY_UP = 38
'CONST KEY_RIGHT = 39
CONST KEY_DOWN = 40
'CONST KEY_SELECT = 41
'CONST KEY_PRINT = 42
'CONST KEY_EXECUTE = 43
'CONST KEY_SNAPSHOT = 44
'CONST KEY_INSERT = 45
'CONST KEY_DELETE = 127        ' Delete key returns 46 in Visual
                                ' Basic for Windows.
'
'CONST KEY_HELP = 47

' KeyCode parameter in KeyDown and KeyUp event procedures
' returns the same value as KeyAscii in the KeyPress event
' procedure for keys corresponding to ASCII printable
' characters (A-Z, a-z, 0-9, ~, [, ], {, }, +, =, etc).  Return
' values will be the ASCII value of the character (see
' ASCII Character Codes topic in Help).  Extended ASCII
' characters can be returned via KeyCode and KeyAscii
' by holding down the ALT key, entering the ASCII number
' of the extended ASCII character, then releasing the
' ALT key.  Note, the NumLock key must be on if the numeric
' keypad is used.

'CONST KEY_NUMPAD0 = 48
'CONST KEY_NUMPAD1 = 49
'CONST KEY_NUMPAD2 = 50
'CONST KEY_NUMPAD3 = 51
'CONST KEY_NUMPAD4 = 52
'CONST KEY_NUMPAD5 = 53
'CONST KEY_NUMPAD6 = 54
'CONST KEY_NUMPAD7 = 55
'CONST KEY_NUMPAD8 = 56
'CONST KEY_NUMPAD9 = 57
'CONST KEY_MULTIPLY = 42
'CONST KEY_ADD = 43
'CONST KEY_SUBTRACT = 45
'CONST KEY_DECIMAL = 46
'CONST KEY_DIVIDE = 47
'CONST KEY_F1 = 112
'CONST KEY_F2 = 113
'CONST KEY_F3 = 114
'CONST KEY_F4 = 115
'CONST KEY_F5 = 116
'CONST KEY_F6 = 117
'CONST KEY_F7 = 118
'CONST KEY_F8 = 119
'CONST KEY_F9 = 120
'CONST KEY_F10 = 121
'CONST KEY_F11 = 122
'CONST KEY_F12 = 123
'CONST KEY_NUMLOCK = 144
'CONST KEY_SCRLOCK = 145

' State (DragOver)

```

```

'CONST ENTER = 0
'CONST LEAVE = 1
'CONST OVER = 2

' -----
' Function parameters
' -----

' MSGBOX parameters
CONST MB_OK = 0
'CONST MB_OKCANCEL = 1
'CONST MB_ABORTRETRYIGNORE = 2
CONST MB_YESNOCANCEL = 3
CONST MB_YESNO = 4
'CONST MB_RETRYCANCEL = 5

CONST MB_DEFBUTTON1 = 0
CONST MB_DEFBUTTON2 = 256
'CONST MB_DEFBUTTON3 = 512

' MSGBOX return values
'CONST IDOK = 1
CONST IDCANCEL = 2
'CONST IDABORT = 3
'CONST IDRETRY = 4
'CONST IDIGNORE = 5
CONST IDYES = 6
CONST IDNO = 7

' -----
' Method parameters
' -----

' DRAG (controls)
'CONST CANCEL_DRAG = 0
'CONST BEGIN_DRAG = 1
'CONST END_DRAG = 2

' SHOW (form)
CONST MODELESS = 0
CONST MODAL = 1

' -----
' Property values
' -----

' Alignment (label)
'CONST LEFT_JUSTIFY = 0           ' 0 - Left Justify
'CONST RIGHT_JUSTIFY = 1          ' 1 - Right Justify
'CONST CENTER = 2                 ' 2 - Center

' BackColor, ForeColor (form, controls)

CONST BLACK = 0
CONST BLUE = 1
'CONST GREEN = 2
CONST CYAN = 3
'CONST RED = 4
CONST MAGENTA = 5
'CONST BROWN = 6
CONST WHITE = 7
'CONST GRAY = 8

```

```

'CONST BRIGHT_BLUE = 9
'CONST BRIGHT_GREEN = 10
CONST BRIGHT_CYAN = 11
'CONST BRIGHT_RED = 12
'CONST PINK = 13
'CONST YELLOW = 14
CONST BRIGHT_WHITE = 15

' BorderStyle (form)
'CONST NONE = 0
'CONST FIXED_SINGLE = 1
'CONST SIZABLE_SINGLE = 2
'CONST FIXED_DOUBLE = 3
'CONST SIZABLE_DOUBLE = 4
'CONST FIXED_SOLID = 5
'CONST SIZABLE_SOLID = 6
                                ' 0 - None
                                ' 1 - Fixed Single
                                ' 2 - Sizable Single
                                ' 3 - Fixed Double
                                ' 4 - Sizable Double
                                ' 5 - Fixed Solid
                                ' 6 - Sizable Solid

' BorderStyle (label, picture box, text box)
'CONST NONE = 0
'CONST SINGLE_LINE = 1
'CONST DOUBLE_LINE = 2
                                ' 0 - None
                                ' 1 - Single Line
                                ' 2 - Double Line
                                ' (label and picture box only)

' DragMode (controls)
'CONST MANUAL = 0
'CONST AUTOMATIC = 1
                                ' 0 - Manual
                                ' 1 - Automatic

' FormType (form - Multiple Document Interface (MDI) vs Single Document
' Interface (SDI) applications)
'CONST NORMAL = 0
                                ' 0 - Normal
                                ' (Normal form in SDI
                                ' applications, child form in
                                ' MDI applications)
'CONST MDI = 1
                                ' 1 - MDI (Container form in
                                ' MDI application)

' MousePointer (form, controls)
'CONST DEFAULT = 0
                                ' 0 - Default (Same
                                ' MousePointer as container
                                ' object's MousePointer)
'CONST BLOCK = 1
'CONST CROSSHAIR = 2
'CONST IBEAM = 3
'CONST ICON = 4
'CONST SIZE_POINTER = 5
'CONST LEFT_ARROW = 6
'CONST SIZE_N_S = 7
                                ' 1 - Block (ASCII 219)
                                ' 2 - Cross (ASCII 197)
                                ' 3 - I-Beam (ASCII 73)
                                ' 4 - Icon (ASCII 002)
                                ' 5 - Size (ASCII 015)
                                ' 6 - Left Arrow (ASCII 027)
                                ' 7 - Size North South
                                ' (ASCII 018)
                                ' 8 - Right Arrow (ASCII 026)
                                ' 9 - Size West East
                                ' (ASCII 029)
'CONST RIGHT_ARROW = 8
'CONST SIZE_W_E = 9
                                ' 10 - Up Arrow (ASCII 024)
                                ' 11 - Hourglass (ASCII 088)
                                ' 12 - Down Arrow (ASCII 025)

' ScrollBar (text box)
'CONST NONE = 0
'CONST HORIZONTAL = 1
'CONST VERTICAL = 2
'CONST BOTH = 3
                                ' 0 - None
                                ' 1 - Horizontal
                                ' 2 - Vertical
                                ' 3 - Both

' Value (check box)
'CONST UNCHECKED = 0
                                ' 0 - Unchecked

```

```

'CONST CHECKED = 1                                ' 1 - Checked
'CONST GRAYED = 2                                ' 2 - Grayed

'WindowState (form)
'CONST NORMAL = 0                                ' 0 - Normal
'CONST MINIMIZED = 1                             ' 1 - Minimized
'CONST MAXIMIZED = 2                            ' 2 - Maximized

' SCREEN.ControlPanel array elements
'CONST ACCESSKEY_FORECOLOR = 0
'CONST ACTIVE_BORDER_BACKCOLOR = 1
'CONST ACTIVE_BORDER_FORECOLOR = 2
'CONST ACTIVE_WINDOW_SHADOW = 3
'CONST COMBUTTON_FORECOLOR = 4
'CONST DESKTOP_BACKCOLOR = 5
'CONST DESKTOP_FORECOLOR = 6
'CONST DESKTOP_PATTERN = 7
'CONST DISABLED_ITEM_FORECOLOR = 8
'CONST MENU_BACKCOLOR = 9
'CONST MENU_FORECOLOR = 10
'CONST MENU_SELECTED_BACKCOLOR = 11
'CONST MENU_SELECTED_FORECOLOR = 12
'CONST SCROLLBAR_BACKCOLOR = 13
'CONST SCROLLBAR_FORECOLOR = 14
'CONST THREE_D = 15
'CONST TITLEBAR_BACKCOLOR = 16
'CONST TITLEBAR_FORECOLOR = 17

'SECPOP90 specific constants.

CONST number_of_segments = 16
CONST max_econ_regions = 99
CONST min_econ_regions = 2
CONST max_number_of_radii = 35
CONST max_county_name_length = 27
CONST record_length = 12
CONST longitude_offset = 91993
CONST latitude_offset = 16610#
CONST number_of_records = 6660337
CONST number_of_counties = 3111
CONST miles_to_kilometers = 1.60934721869#
CONST kilometers_to_miles = 0.62136994950#
CONST max_layers = 10
CONST seconds_in_a_day = 86400
CONST pi = 3.14159265358979#

```

**FILENAME: COMMON.BI**

```
COMMON SHARED leave_calculation AS INTEGER
COMMON SHARED rec AS STRING * record_length
COMMON SHARED xlon AS SINGLE, ylat AS SINGLE
COMMON SHARED 1pop AS INTEGER, area AS LONG, county_code AS INTEGER
COMMON SHARED 1dir AS INTEGER, 1rad AS INTEGER, 1byte AS LONG

COMMON SHARED number_of_radii AS INTEGER, numrad AS INTEGER
COMMON SHARED Population_multiplier AS SINGLE, scale AS SINGLE
COMMON SHARED radial_distance() AS SINGLE, raddis() AS SINGLE
COMMON SHARED regional_radii() AS SINGLE
COMMON SHARED population() AS long
COMMON SHARED population_threshold() AS long
COMMON SHARED slon AS SINGLE, slat AS SINGLE
COMMON SHARED dismax AS SINGLE, sitid$
COMMON SHARED dpdlat AS SINGLE, dpdlon AS SINGLE, sdpdla AS SINGLE
COMMON SHARED bndryw AS SINGLE, bndrye AS SINGLE
COMMON SHARED bndryn AS SINGLE, bndrys AS SINGLE
COMMON SHARED aminla AS SINGLE, dgtord AS SINGLE

COMMON SHARED region_index() AS INTEGER, number_econ_regions AS INTEGER
COMMON SHARED econ_data() AS economic_data
COMMON SHARED table_type AS INTEGER
COMMON SHARED x_position AS INTEGER, y_position AS INTEGER

COMMON SHARED sector_population() AS LONG
COMMON SHARED sector_area() AS LONG
COMMON SHARED sector_frcInd() AS SINGLE

COMMON SHARED county_state() AS STRING * 2
COMMON SHARED county_name() AS STRING * max_county_name_length
COMMON SHARED county_frcInd() AS SINGLE, county_frmfrc() AS SINGLE
COMMON SHARED county_dpf() AS SINGLE, county_asfp() AS SINGLE
COMMON SHARED county_vfrm() AS SINGLE, county_vnfrm() AS SINGLE

COMMON SHARED directions() AS STRING

COMMON SHARED layer_number() AS INTEGER
COMMON SHARED data_flag() AS INTEGER
COMMON SHARED display_flag() AS INTEGER
COMMON SHARED active_layer() AS INTEGER
COMMON SHARED layer_name() AS STRING
COMMON SHARED pts_color() AS INTEGER
COMMON SHARED pts_type() AS INTEGER
COMMON SHARED pts_size() AS INTEGER
COMMON SHARED pts_mode() AS INTEGER
COMMON SHARED lines_color() AS INTEGER
COMMON SHARED lines_type() AS INTEGER
COMMON SHARED lines_size() AS INTEGER
COMMON SHARED lines_mode() AS INTEGER
COMMON SHARED polyg_color() AS INTEGER
COMMON SHARED polyg_type() AS INTEGER
COMMON SHARED polyg_size() AS INTEGER
COMMON SHARED polyg_mode() AS INTEGER
COMMON SHARED text_color() AS INTEGER
COMMON SHARED text_type() AS INTEGER
COMMON SHARED text_size() AS INTEGER
COMMON SHARED text_mode() AS INTEGER

COMMON SHARED radial_area() AS SINGLE
```

**FILENAME: SECPOP90.MAK**

SECPOP90.BAS  
CALCCODE.BAS  
CMNDLG.BAS  
OUTPCODE.BAS  
PROBCODE.BAS  
ROSETTE.BAS  
RPTABLE1.BAS  
SITECODE.BAS  
UTILITY.BAS

CALCFORM.FRM  
CIRCLE.FRM  
CMNDLGF.FRM  
DENSFORM.FRM  
DISCLAIM.FRM  
MAIN.FRM  
OUTPFORM.FRM  
PROBFORM.FRM  
REGION.FRM  
SETUP.FRM  
SITEFORM.FRM  
TABLE1.FRM

**FILENAME: MAKEFILE**

```
# -----
# Makefile:      makefile
#
# -----
#
# Author(s):    Steven Humphreys
#                 Sandia National Laboratories
#                 Accident Analysis / Consequence Assessment Department
#                 Albuquerque, NM 87185-0748
#                 (505) 844-7223
#
# Description: Makefile for creating SECPOP90 executable.
#
# Usage:        Type make at the command line.
#
# Inputs:
#
# Outputs:
#
# Variables:
#
# Calls:
#
# Assumptions: A Lahey or compatible make facility.
#
# Version:      1.0
#
# Date:         June 17, 1996
#
# Revision:
# History:     None - initial version.
#
# -----
.
.SUFFIXES:      .obj .bas .frm

BC =           c:\vbdos\bc.exe
BFLAGS =        /Ah /D /E /G3 /O /X

LINKER =        c:\vbdos\link.exe
LFLAGS =        /E /F /PACKC /PACKF /SE:256
LIBRARY =       c:\vbdos\lib\vbdcl10e.lib c:\vbdos\lib\vbdos.lib

PROGRAM =       c:\secpop90\secpop90.exe

BSRCS =         secpop90.bas\
                calccode.bas cmandlg.bas outpcode.bas probcode.bas\
                rosette.bas rptable1.bas sitecode.bas utility.bas

FSRCS =         calcform.frm circle.frm cmndlgf.frm densform.frm\
                disclaim.frm main.frm outpform.frm probform.frm\
                region.frm setup.frm siteform.frm table1.frm

OBJECTS =       $(BSRCS,.bas=.obj) $(FSRCS,.frm=.obj)

.bas.obj:
        $(BC) $(BFLAGS) $< $@ nul.lst

.frm.obj:
        $(BC) $(BFLAGS) $< $@ nul.lst
```

```

$(PROGRAM):      $(OBJECTS)
                  $(LINKER) $(LFLAGS) $(OBJECTS), $@, nul.map, \
                  $(LIBRARY), nul.def

# ----

# Basic modules include file dependencies

calccode.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
cmndlg.obj:    secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
outpcode.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
main.obj:       secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
probcode.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
rosette.obj:   secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
rptable1.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
sitecode.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
utility.obj:   secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi

calcform.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
circle.obj:    secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
cmndlgf.obj:   secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
densform.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
disclaim.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
outpform.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
probform.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
region.obj:   secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
secpop90.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
setup.obj:     secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
siteform.obj:  secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi
table1.obj:   secpop90.bi types.bi declare.bi const.bi common.bi \
                forms.bi

# -----

```

**FILENAME: MP\_SETUP.BAS**

```
CHDIR "mapplan"
SHELL "setup.exe"
CHDIR ".."
RUN "secpop90"
```

**FILENAME: MP\_SETUP.MAK**

MP\_SETUP.BAS

**FILENAME: RUN\_MPPR.BAS**

```
CHDIR "mapplan"
SHELL "mppr.exe . map_out.spc"
CHDIR ".."
RUN "secpop90"
```

**FILENAME: RUN\_MPPR.MAK**

RUN\_MPPR.BAS



