

Creating a User-Friendly Menu System without SAS/AF®*

Robert F. Davis
Calspan Corporation/AEDC Operations
Arnold Engineering Development Center
Arnold Air Force Base, TN 37389

ABSTRACT

Because of budget constraints, many companies have chosen not to license the SAS® application facility (SAS/AF) product. This makes creating a user-friendly menu system difficult. A reasonable facsimile of a user-friendly menu system can be created for most applications using Window Display, Macro Language, PROC PMENU in SAS/BASE®, PROC FSEDIT with Screen Control Language from SAS/FSP®, and SAS/SHARE®. However, this system will run slower than a similar system created with SAS/AF.

PROC PMENU is used as the basis for this system and creates the main and submenu definitions. Data steps and PROC FSEDIT with screen control language are called from the main or submenus to augment and enhance PROC PMENU's capabilities. Macro Language is also used to simplify the menu process. Through special techniques, arguments from menu items selected by the user are passed to an application program for processing, resulting in the final output of the application. The system is set up to be shared by multiple users.

The intended audience for this paper should have a good understanding of SAS data steps, PROC PMENU, simple Macro Language elements, PROC FSEDIT, screen control language under PROC FSEDIT, and SAS/SHARE. The ideas and examples in this paper are meant to be a guideline for creating a user-friendly menu system without using SAS/AF. It should inspire creativity to develop a similar system around your own applications and needs. This paper presents a simplified alternative to SAS/AF. SAS/AF is an excellent product and can do much more than is presented as well as being faster in development and run time.

INTRODUCTION

The original system on which this paper is based was developed for the Calspan Planning and Work Control Center. The system allows engineers and managers unfamiliar with SAS to get data reports from a long-range planning database. The user enters a TSO execute command to bring up the menu system. After the main menu is shown on the screen, the user marks menu items or fills in blanks to get the required report. There is also a menu item to allow the user to send the report to any remote printer connected to the current system.

A menu tree and cross reference is presented in Fig. 1. Above each block in parentheses is the type of menu. To the left is a number which corresponds to a number in the descriptive text. To the right of each block are references to figures that apply to that subject.

*The research reported herein was performed by the Arnold Engineering Development Center (AEDC), Air Force Materiel Command. Work and analysis for this research were done by personnel of Calspan Corporation/AEDC Division, technical services contractor for the AEDC aerospace flight dynamics facilities. Further reproduction is authorized to satisfy needs of the U. S. Government.

THE SYSTEM IN GENERAL

The example system in this discussion is comprised of six different menus or windows, one containing two screens. The menu system begins with the command list presented in Fig. 2. In this figure, libname "LIB" is allocated to the common source library, and the libname "USERLIB" is assigned to the common SASUSER library. Both are shared so multiple users can access the system at the same time. RSASUSER SAS option is defined in Fig. 2, so users cannot modify function key definitions. Preset function key definitions are shown in Fig. 3. Because this system may be used by non-SAS users, the designer may want to restrict PF key definitions to the suggested PF1 through PF12. The system may be upset if screens are accessed out of order. The author uses PF13 through PF24 for debugging purposes only.

(1) The Main Menu Window

The autoexec file called by the command list in Fig. 2 is the code for main menu window definition (Fig. 4). This window definition uses a PMENU catalog entry to display a PMENU in place of the command line at the top of the screen. The window definition itself describes the PMENU entries and gives the user instructions at the bottom of the screen. Looking at Fig. 4 we see an assignment for the menu library where the PMENU catalog entry is stored. Then the current fiscal year is calculated and stored in global common memory for use later by the menus and programs as a default starting year. The main menu window is now defined in a data step with a window statement using a PMENU catalog entry. Text follows, describing each PMENU entry and the instructions to be presented to the user. Display manager commands are now executed to turn on PMENUS for all windows and turn off autopop for the output window so the output window does not come up while the reports are being processed. When the code in Fig. 4 is executed, the window with the PMENU shown in Fig. 5 is presented to the user. The main PMENU and submenus are generated from PROC PMENU and stored in a SAS catalog entry. Figure 6 is the code for the PROC PMENU of this system. Because PROC PMENU executes outside the menu system and builds a catalog entry for use by the menu system, the fiscal year calculation has to be redefined in the PROC PMENU code (Fig. 6). The main PMENU is declared in the first menu statement. Three item statements define the PMENU items, "REPORTS," "PRINT," and "EXIT."

(2) Selection list (REPORTS)

As shown in Fig. 6 (PMENU source code), the item statement REPORTS in the main menu is directed to a menu statement named reports. This menu statement defines a selection list of

five items (single page menu, multi-page menu, tso batch job, cancel, and exit). The window created by this menu statement is pictured in Fig. 7.

(3) Single Page Menu

The first item in the submenu under main menu REPORTS is SINGLE PAGE MENU. This item is directed to a dialog statement named SINGLE. This dialog statement consists of a command list and various selection criteria formed with checkbox and radio-box statements. The window created by this dialog is pictured in Fig. 8. After the user selects "OK" for this dialog box, the command line is executed with the user selections replacing the original arguments. This command list starts with a Macro let statement that stores the original argument list in global common memory at variable ARGLIST. The command list then ends the PMENU, makes the program editor window the current window, and submits a macro from the common source library named PRJLIST for execution. The syntax of ARGLIST is critical because the SAS macro function SCAN is used to parse the argument list and store each argument into separate global common variables. There must be a leading period and space before the first argument and a period and space after each argument. A period or space cannot be used in the arguments. These global variables are then used to select records from a data set and create a report. As shown in Fig. 9, we find the source code for the MACRO PRJLIST. First is a group of %LET statements using the %SCAN function to parse the variable ARGLIST. Next, a SAS data set containing the information for this report is assigned. Then the printer page size is declared with an options statement. The %IF statement declares the current fiscal year as default start year if the user has not specified one. Next, observations from the input data set are selected, matching the global common arguments against the data set variables. The code that follows is unique to creating this report. The global variable PRY_TYPE from the original argument list is used to make a complete local variable name. The last step is to recall the main menu window. This is done by using a %INCLUDE statement to include the main menu window code for execution.

(4) Multi-Page Menu

The second item in the submenu under main menu REPORTS is MULTI PAGE MENU. Referring to PROC PMENU source (Fig. 6), this item statement is directed to a selection statement named MULTI that causes a display manager command line to be executed. The command line ends the PMENU, selects the program editor window, and submits a macro (LRPMENU) from the common source library with an argument naming a report program to be called at the end of the macro LRPMENU. This selection is complicated. There is no way for PROC PMENU to do multi-page menus. LRPMENU is a macro (Fig. 10) that presents a way of doing multi-page menus. LRPMENU starts by declaring some global variables and then turns all PMENUS off. A libname statement assigns the data set from which data for a report will be selected. A data step with a list of attribute statements defines all the variables that will be in the menu screens. A "%IF" statement checks to see if this same program is being run again. If it is, the set statement is executed to keep the previous menu selections. PROC FSEDIT is now invoked using the variables from the previously defined data step and some predefined screens. The screen definitions (Figs. 11, 12) stored in a SAS screen catalog entry also contain screen control language (Fig. 13). The FSEINIT section of this screen control language starts by protecting the program name field, clearing the abort flag, and setting the control parameter to always enter the main label when

the user presses the enter key. The INIT section fills in the program, name from the global variable passed from the PMENU command list. If the last program run is not the current program the variable START_FY is set to the current fiscal year. The MAIN: section of this screen control language gets the command from the command line and looks to see if it is an abort. If it is, the abort flag is set and the FSEDIT session is ended by executing an end command upon return from the main routine. In the TERM: section, if the abort flag has not been set, some validity checks are made on the menu variables. If these checks fail, an error message is presented under the command line on the menu screen indicating what error was detected. If otherwise the FSETERM: section stores the current program name as the last program and exits the FSEDIT session returning program control to LRPMENU. PF keys are defined for the FSEDIT screen to control user actions and are described at the bottom of the screens. If the user chooses the abort PF key, the FSEDIT screen control language sets a global flag. After termination of the FSEDIT session, this flag is checked by a %IF statement. If the abort flag is set, the code following the %IF statement is skipped and the main menu window is recalled. If the user chooses the PF key to continue, the select criteria from screen one in the form of a data set is used to select observations from a SAS data set.

Next is an optional step. There was a requirement of this system for the user to be able to input up to two lines of title of his choosing and to format on a separate line the items the user marked as selection criteria. The macro LRPTITLE (Fig. 14) was developed to process this title information. The third step is to format the sort select criteria from FSEDIT screen two into a global variable named BY for use by the report program as a by statement for a PROC SORT. The last step uses a %INCLUDE statement to pull in and execute the source for the report program from the common source library with the argument that was passed by the PMENU call to macro LRPMENU being the source name for the report program (Fig. 15). Note in Fig. 15 the use of the global variable &BY in the PROC SORT. At the end of the report program, a %INCLUDE statement recalls the main menu window.

(5) Batch Submit Verification

The third item in the submenu under main menu REPORTS is TSO BATCH JOB. This item statement (Fig. 6) is directed to a dialog statement BATCH which presents a verification menu (Fig. 16.) If "OK" is selected, a TSO batch job (Fig. 17) is executed. Only one rbutton is defined in this menu and it is default. Therefore the user will have a chance to abort this menu, and it is default. Therefore, the user will have a chance to abort this menu by selecting cancel after reading the informational message presented in the menu. Because the dialog command line did not execute an end statement, Pmenu is not terminated but returns to the previous menu.

(6) Reports Window Cancel

The fourth item in the submenu under main menu REPORTS is CANCEL. This item statement (Fig. 6) contains a display manager command that returns to the previous menu.

(7) Reports Window Exit

The fifth item in the submenu under main menu REPORTS is EXIT. This item is directed to a selection statement called exit. Selection exit executes a display manager bye command and exits the SAS session.

(8) Print Menu

The main menu selection PRINT is directed to a dialog statement called print. Dialog print allows the user to select a remote printer, number of copies to print, and gives the user instructions. The window presented by this dialog is shown in Fig. 18. Upon selecting "OK" at the bottom of this submenu, the user executes the command line associated with the dialog print. This command line has six display manager commands. "END;"; terminates the user window. "PGM;"; selects the program editor window as the current window. "INC LIB(PRINT);" includes from the common source code library a member named print into the program editor window. "C X1 @1; C X2 @2;" commands change the first occurrence of strings "X1" and "X2" in the program editor window with the remote printer assignment and number of copies respectively. Finally, the "SUBMIT;" command executes the modified source code from the program editor window. Figure 19 shows the original source code from the common source library.

(9) Main Menu Exit

The main menu selection "EXIT" is directed to a selection statement named exit. Selection exit executes a display manager command bye and exits the SAS session. This is a different statement than a similar one for the reports selection list item exit because Proc Pmenu requires a complete tree for each item declared.

SUMMARY

The menu system presented here is a reasonable facsimile of a user-friendly menu system for obtaining data reports. It is easy enough for people with little or no knowledge of the SAS system, to use. The code examples used in this paper were extracted and modified from an actual working system at Calspan. The extractions and modifications were used to build a separate system for this paper. All examples presented here have been tested and work the same as the original. The system is running on an IBM® 3090 mainframe under TSO with SAS version 6.08.

REFERENCES

SAS Language Reference, Version 6 First Edition.

SAS Procedures Guide, Version 6 Third Edition.

SAS Guide to Macro Processing, Version 6 Second Edition.

SAS Technical Report P-222, Changes and Enhancements to Base SAS Software, Release 6.07.

SAS Technical Report P-242, SAS Software: Changes and Enhancements, Release 6.08.

SAS/FSP Software Usage and Reference, Version 6, First Edition.

SAS Screen Control Language Usage, Version 6, First Edition.

SAS Screen Control Language Reference, Version 6, First Edition.

SAS Technical Report P-199 Using SAS Screen Control Language in Release 6.06.

SAS Technical Report P-216 SAS/AF Software, SAS/FSP software, and SAS Screen Control Language: Changes and Enhancements, Release 6.07.

SAS/SHARE Usage and Reference, Version 6, First Edition.

SAS Technical Report P-212 SAS/SHARE Software for the MVS Environment, Release 6.07.

SAS Companion for the MVS Environment, Version 6, First Edition.

SAS Technical Report P-218 Changes and Enhancements to the SAS System for the MVS Environment Release 6.07.

ACKNOWLEDGMENTS

SAS is a registered trademark of SAS Institute Inc. SAS Campus Drive, Cary, N.C. 27513

Many thanks to Theresa Cates of OAO Corp. and the SAS technical support people for help in various areas of development of this system.

AUTHOR CONTACT

Calspan FSD Engineering Bldg.
430 Second Street
Arnold AFB, TN 37389-4200
ATTN: Robert F. Davis
(615) 454-5924

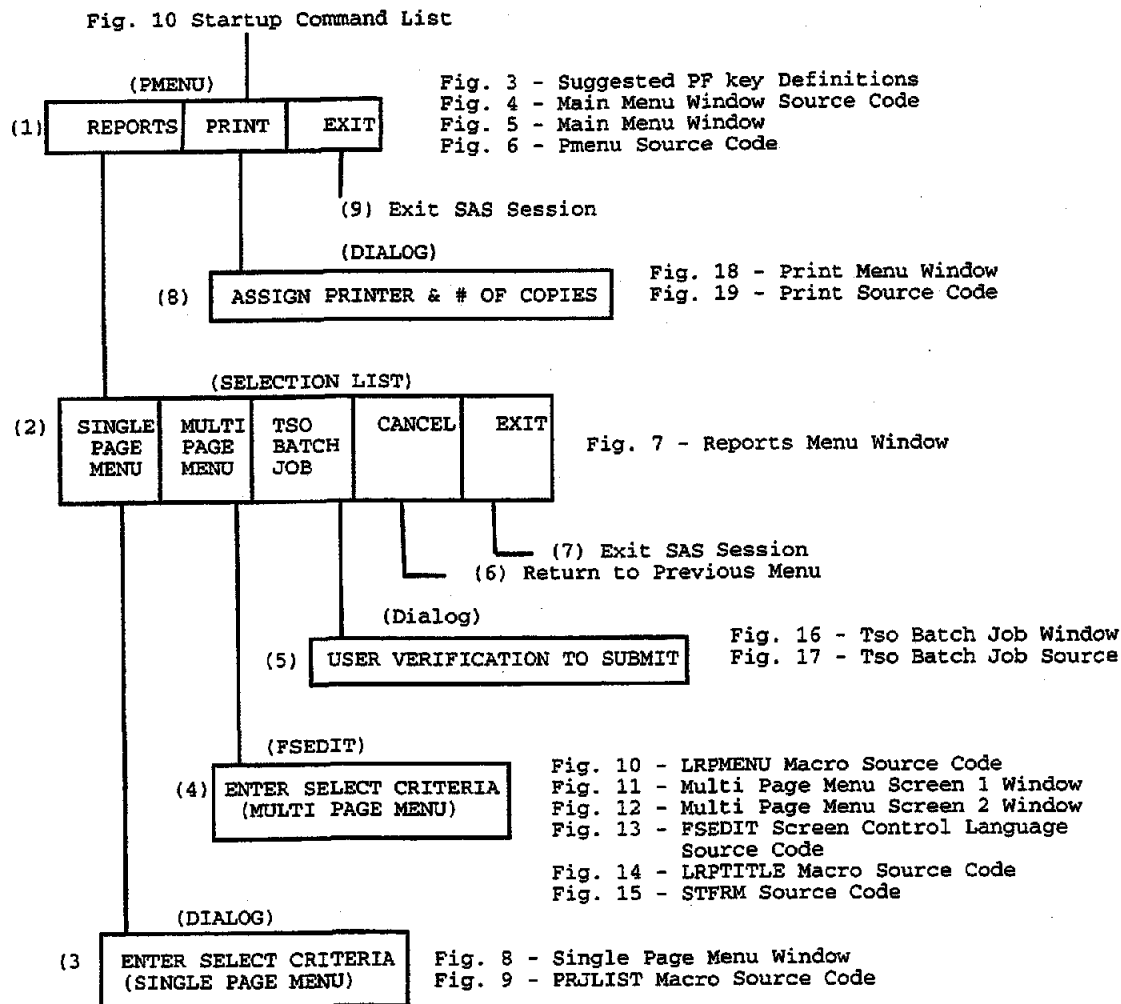


Fig. 1. MENU TREE AND CROSS REFERENCE.

```

1  /*****
1  /* COMPANY: CALSPAN/AEDC
1  /* SOURCE CODE NAME: 'CAL.SEUG.MENU'
1  /* AUTHOR: ROBERT F. DAVIS
1  /* PURPOSE:
1  /*   EXAMPLE CODE FOR USE AT SAS SOUTHEASTERN USE GROUP.
1  /*   THIS IS A TSO CLIST THAT STARTS THE CALSPAN PLANNING AND
1  /*   WORK CONTROL MENU SYSTEM.
1  /*****
1  ALLOC F(LIB) DA('CAL.SEUG.LIB') SHR
1  ALLOC F(SEUGUSER) DA('CAL.SASUSER.LIB') SHR
1  SAS6 OPTIONS -
1  ('MACRO MACROGEN NOMPRINT SYMBOLGEN MLOGIC RSASUSER -
1  SASUSER-SEUGUSER') -
1  AUTOEXEC(''CAL.SEUG.LIB(SEUGWIN)''')
1  FREE F(LIB)
1  FREE F(SEUGUSER)

```

00000300
00000500
00000700
00000900
00001100
00001200
00001300
00001400
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500

Fig. 2. STARTUP COMMAND LIST

KEY	DEFINITION	USAGE DESCRIPTION
F1		
F2		
F3		
F4		
F5	out	View Output
F6	pgm;next	Return from View Output
F7	backward	Page Output Backward
F8	forward	Page Output Forward
F9		
F10	left	Scroll Output Left
F11	right	Scroll Output Right
F12	home	Position Cursor to Pmenu or Command Line

Fig. 3. Suggested Pmenu Function Key Definitions.

KEY	DEFINITION	USAGE DESCRIPTION
F13	help	SAS Help
F14	zoom	Toggle Zoom
F15	end	Terminate SAS session
F16		
F17	rfind	Repeat Find
F18	rchange	Repeat Change
F19	prevwind	Go To Previous Window
F20	next	Go To Next Window
F21	pmenu	Toggle Pmenus
F22	command	Toggles Pmenu in the Active Window
F23		
F24	home	Position Cursor to Pmenu

Fig. 3. Concluded.

```

1  /***** */ 00030038 *
1  /* COMPANY: CALSPAN/AEDC */ 00050038 *
1  /* SOURCE CODE NAME: SEUGWIN */ 00070038 *
1  /* AUTHOR: ROBERT F. DAVIS */ 00090038 *
1  /* PURPOSE: */ 00110038 *
1  /* EXAMPLE WINDOW CODE FROM CALSPAN PLANNING AND WORK CONTROL */ 00120038 *
1  /* MENU SYSTEM FOR USE AT SAS SOUTHEASTERN USERS GROUP. */ 00130038 *
1  /***** */ 00170038 *
1  OPTIONS MAUTOSOURCE SASAUTOS=LIB; 00180038 *
2  OPTIONS MRECALL; 00190038 *
1  /***** */ 00210038 *
1  /* ASSIGN THE MENU LIBRARY */ 00220038 *
1  /***** */ 00230038 *
1  LIBNAME SEUG 'CAL.SEUG.MENU.LIB' SERVER=SERVERS; 00240038 *
1  /***** */ 00260038 *
1  /* CALCULATE AND STORE CURRENT FISCAL YEAR */ 00270038 *
1  /***** */ 00280038 *
1  %GLOBAL CUR_FY; 00300038 *
1  DATA DATE; 00310038 *
4  CUR_FY=YEAR(TODAY())+INT(MONTH(TODAY())/10); 00320038 *
4  CALL SYMPUT('CUR_FY',PUT(LEFT(CUR_FY),$4.)); 00330038 *
1  RUN; 00340038 *
1  DATA SEUG_WIN; 00350038 *
1  /***** */ 00360038 *
1  /* NAME THE MAIN MENU WINDOW AND ASSIGN A PMENU CATALOG ENTRY */ 00370038 *
1  /***** */ 00380038 *
1  WINDOW SEUGWIN MENU=SEUG.MENU.SEUGMENU 00390038 *
1  /***** */ 00400038 *
1  /* DEFINE THE MAIN MENU WINDOW */ 00410038 *
1  /***** */ 00420038 *
4  #1 @25 'CALSPAN WORK CONTROL MAIN MENU' 00430038 *
4  #4 @8 'REPORTS - CREATE REPORTS' 00440038 *
4  #5 @8 'PRINT - PRINT OUTPUT(S)' 00450038 *
4  #6 @8 'EXIT - EXIT THE WORK CONTROL CENTER SYSTEM' 00460038 *
4  #10 @5 'USE TAB KEY TO MAKE SELECTION FROM MENU BAR ABOVE' 00470038 *
4  #11 @5 'AND PRESS ENTER.' 00480038 *
4  #19 @1 'F5=VIEW OUTPUT. F6=RETURN FROM VIEW OUTPUT.' 00490038 *
4  #19 @50 'F7=PAGE BACKWARD.' 00500038 *
4  #20 @1 'F8=PAGE FORWARD F10=PAGE LEFT. F11=PAGE RIGHT.'; 00510038 *
4  DM 'PMENU ON'; /* TURN PMENUS FOR ALL WINDOWS ON */ 00520038 *
4  DM OUT 'AUTOPOP OFF'; /* STOP OUTPUT WINDOW FROM POPPING UP */ 00530038 *
1  /***** */ 00540038 *
1  /* DISPLAY THE MAIN MENU */ 00550038 *
1  /***** */ 00560038 *
1  DISPLAY SEUGWIN; 00570038 *
1  00580038 *
1  00590038 *
1  00600038 *
1  RUN; 00610038 *

```

Fig. 4. MAIN MENU WINDOW SOURCE CODE.

```

SEUGWIN
REPORTS | PRINT |... EXIT |

CALSPAN WORK CONTROL MAIN MENU

REPORTS - CREATE REPORTS
PRINT - PRINT OUTPUT(S)
EXIT - EXIT THE WORK CONTROL CENTER SYSTEM

USE TAB KEY TO MAKE SELECTION FROM MENU BAR ABOVE
AND PRESS ENTER.

F5=VIEW OUTPUT. F6=RETURN FROM VIEW OUTPUT. F7=PAGE BACKWARD.
F8=PAGE FORWARD F10=PAGE LEFT. F11=PAGE RIGHT.

```

Fig. 5. MAIN MENU WINDOW

```

1  /*****
1  /* COMPANY: CALSPAN/AEDC
1  /* SOURCE CODE NAME: SEUGMENU
1  /* AUTHOR: ROBERT F. DAVIS
1  /* PURPOSE:
1  /* EXAMPLE PROC PMENU CODE FROM CALSPAN PLANNING AND WORK CONTROL
1  /* MENU SYSTEM FOR USE AT SAS SOUTHEASTERN USERS GROUP.
1  /*****
1  LIBNAME SEUG 'CAL.SEUG.MENU.LIB';
1  /*****
1  /* CALCULATE THE CURRENT FISCAL YEAR
1  /*****
1  %GLOBAL CUR_FY;
1  DATA CUR_FY;
4  CUR_FY=YEAR(TODAY())+INT(MONTH(TODAY())/10);
4  CALL SYMPUT('CUR_FY',PUT(LEFT(CUR_FY),$4.));
1  RUN;

1  PROC PMENU CAT=SEUG.MENU;
1  /*****
1  /* DEFINE THE MAIN PMENU
1  /*****
4  MENU SEUGMENU;
7  ITEM 'REPORTS |' MENU=REPORTS;
7  ITEM 'PRINT |' DIALOG=PRINT;
7  ITEM 'EXIT |' SELECTION=EXIT;

1  /*****
1  /* SUBMENU FOR MAIN MENU ITEM REPORTS
1  /*****
4  MENU REPORTS;
7  ITEM 'SINGLE PAGE MENU' DIALOG=SINGLE;
7  ITEM 'MULTI PAGE MENU' SELECTION=MULTI;
7  ITEM 'TSO BATCH JOB' DIALOG=BATCH;
7  ITEM CANCEL;
7  ITEM 'EXIT' SELECTION=EXIT;

1  /*****
1  /* EXAMPLE OF HOW TO NULLIFY A MENU ITEM
1  /*****
4  /*
7  MENU SINGLE;
7  ITEM 'THIS REPORT IS BEING WORKED ON' SELECTION=RETURN;
7  SELECTION RETURN '';
4  */

1  /*****
1  /* SUB SUBMENU TO MAIN MENU ITEM REPORTS FOR SINGLE PAGE ITEM
1  /*****
1  DIALOG SINGLE
1  %LET ARGLIST= . @1. &1. &2. &3. &4. &5. &6. &7. &8. &9. &10. &11.
1  &12. &13. &14. &15. &16. &17. &18. &19. &20. &21. &22. %1. ;END;PGM;
1  SUB '%PRJLIST';;

1  /*****
1  /* ALTERNATE MEANS OF SUBMITTING THE ABOVE FOR PROCESS
1  /* THE METHOD BELOW HAS A CHARACTER LIMIT THAT RESTRICTS THE
1  /* MACRO CALL. FOR LONG LISTS OF ARGUMENTS THE ABOVE IS PREFERRED.*/
1  /*****
1  /*
1  'END;PGM;SUB '%PRJ_LIST(@1,&1,&2,&3,&4,&5,&6,&7,&8,&9,&10,&11,&12,
1  &13,&14,&15,&16,&17,&18,&19,&20,&21,&22,%1)'';;
1  */
6  TEXT #1 @15 'LONG RANGE PLANNING SELECTION CRITERIA';
6  TEXT #2 @5 'ENTER STARTING FY (DEFAULT=&CUR_FY)';
6  TEXT #2 @40 LEN=4;

9  TEXT #3 @5 'FUNDING CATEGORY: ';
9  CHECKBOX #3 @22 'DBA';
9  CHECKBOX #3 @28 'RBA';
9  CHECKBOX #3 @34 'DP';
9  CHECKBOX #3 @40 'ACT';
9  TEXT #4 @5 'PROGRAMS: ';
9  CHECKBOX #5 @5 'TTS';
9  CHECKBOX #5 @11 'TEC';
9  CHECKBOX #5 @17 'M&R';
9  CHECKBOX #5 @23 'I&M';
9  CHECKBOX #5 @29 'A&E';
9  CHECKBOX #5 @35 'ADP';
9  CHECKBOX #5 @41 'MCP';
9  CHECKBOX #5 @47 'ENV';
9  CHECKBOX #5 @53 'SSI';
9  CHECKBOX #5 @59 'SEC';
9  CHECKBOX #5 @65 'SUP';
9  TEXT #6 @5 'COMPLEX(S): ';
9  CHECKBOX #6 @17 'ARC';
9  CHECKBOX #6 @23 'CHM';
9  CHECKBOX #6 @29 'HYP';
9  CHECKBOX #6 @35 'MUL';
9  CHECKBOX #6 @41 'PLT';
9  CHECKBOX #6 @47 'RGS';
9  CHECKBOX #6 @53 'TRN';
9  TEXT #7 @5 'SELECT PRIORITY TYPE';
9  RADIOBOX DEFAULT=4;
9  RBUTTON #7 @25 'PRYX' SUBSTITUTE='X';

```

Fig. 6 PMENU SOURCE CODE

```

9      REBUTTON #7 @32 'PRYY' SUBSTITUTE='Y';
9      REBUTTON #7 @39 'PRYY' SUBSTITUTE='YY';
9      REBUTTON #7 @46 'PRYZ' SUBSTITUTE='Z';

1  /*****
1  /* SUB SUBMENU TO MAIN MENU ITEM REPORTS FOR MULTI PAGE ITEM */
1  /*****
4  SELECTION MULTI 'END;PGM;SUB '%LRPMENU(STFRM)'';

1  /*****
1  /* SUB SUBMENU TO MAIN MENU ITEM REPORTS FOR
1  /* TSO BATCH JOB ITEM
1  /*****
4  DIALOG BATCH
9  'S1';
7  TEXT #1 @3 'IF SELECTED A BATCH JOB WILL BE SUBMITTED';
7  TEXT #2 @3 'FOR RUN AFTER 1600 HOURS.';
10 RADIOBOX DEFAULT=1;
10 REBUTTON #4 @15 'RUN MASTER'
11 SUBSTITUTE='TSO SUBMIT 'CAL.SEUG.LIB(TPRBAT)';

1  /*****
1  /* SUBMENU TO MAIN MENU ITEM REPORTS FOR EXIT ITEM
1  /*****
7  SELECTION EXIT 'BYE';

1  /*****
1  /* SUBMENU TO MAIN MENU ITEM PRINT
1  /*****
7  DIALOG PRINT
12 'END;PGM;INC LIB(PRINT);C X1 @1;C X2 @2;SUBMIT';

7  TEXT #1 @3 'ENTER REMOTE PRINTER DESTINATION';
7  TEXT #3 @10 'DEFAULT = RMT107';
7  TEXT #3 @30 LEN=6;
7  TEXT #5 @3 'ENTER NUMBER OF COPIES (1-99)';
7  TEXT #6 @10 'DEFAULT = 1';
7  TEXT #6 @26 LEN=2;
7  TEXT #13 @10
12 'PLEASE MAKE SELECTIONS USING THE TAB AND ENTER KEY.';
7  TEXT #14 @10 'WHEN ALL SELECTIONS HAVE BEEN MADE.';
7  TEXT #15 @10 'SELECT OK TO CONTINUE OR CANCEL TO ABORT.';

1  /*****
1  /* MAIN MENU ITEM EXIT
1  /*****
7  SELECTION EXIT 'BYE';

1  RUN;
1  QUIT;

```

01060019 *
01070019 *
01080019 *
01090019 *
01100019 *
01110019 *
01120019 *
01130019 *
01140019 *
01150019 *
01160019 *
01170019 *
01180019 *
01190019 *
01200019 *
01210019 *
01220019 *
01230019 *
01240019 *
01250019 *
01260019 *
01270019 *
01280019 *
01290019 *
01300019 *
01310019 *
01320019 *
01330019 *
01340019 *
01350019 *
01360019 *
01370019 *
01380019 *
01390019 *
01400019 *
01410019 *
01420019 *
01430019 *
01440019 *
01450019 *
01460019 *
01470019 *
01480019 *
01490019 *
01500019 *
01510019 *
01520019 *
01530019 *
01540019 *
01550019 *

Fig. 6. CONCLUDED

SEUGWIN	
REPORTS PRINT ... EXIT SINGLE PAGE MENU... MULTI PAGE MENU TSO BATCH JOB... CANCEL EXIT	CALSPAN WORK CONTROL MAIN MENU E REPORTS UTPUT(S) WORK CONTROL CENTER SYSTEM USE TAB KEY TO MAKE SELECTION FROM MENU BAR ABOVE AND PRESS ENTER. F5=VIEW OUTPUT. F6=RETURN FROM VIEW OUTPUT. F7=PAGE BACKWARD. F8=PAGE FORWARD F10=PAGE LEFT. F11=PAGE RIGHT.

Fig. 7. REPORTS MENU WINDOW.

SEUGWIN	
REPORTS PRINT ... EXIT SI SINGLE PAGE MENU... MU LONG RANGE PLANNING SELECTION CRITERIA TS ENTER STARTING FY (DEFAULT=1994) CA FUNDING CATEGORY [[DBA] [[RBA] [[DP] [[ACT EX PROGRAMS: [[TTS] [[TEC] [[M&R] [[I&M] [[A&E] [[ADP] [[MCP] [[ENV] [[SSI] [[SEC] [[SUP COMPLEX(ES): [[ARC] [[CHM] [[HYP] [[MUL] [[PLT] [[RGS] [[TRN SELECT PRIORITY TYPE o PRYX o PRYY o PRYY . PRYZ [OK] [Cancel]	F5=VIEW OUTPUT. F6=RETURN FROM VIEW OUTPUT. F7=PAGE BACKWARD. F8=PAGE FORWARD F10=PAGE LEFT. F11=PAGE RIGHT.

Fig. 8. SINGLE PAGE MENU WINDOW.

```

1  /*****
1  /* COMPANY: CALSPAN/AEDC
1  /* SOURCE CODE NAME: PRJLIST
1  /* AUTHOR: ROBERT F. DAVIS
1  /* PURPOSE:
1  /* EXAMPLE CODE FOR USE AT SAS SOUTHEASTERN USER GROUP.
1  /* THIS CODE CREATES A PROJECTS LIST REPORT FROM THE
1  /* CALSPAN PLANNING AND WORK CONTROL LONG RANGE PLANNING DATA
1  /* BASE WITH INFORMATION SUPPLIED FROM THE MENU SYSTEM.
1  /*****
1  %MACRO PRJLIST;
1  /*****
1  /* PARSE THE INPUT PARAMETERS AND STORE IN GLOBAL COMMON VARIABLES */
1  /*****
1  %LET START_FY=%SCAN(&ARGLIST,1,%STR());
1  %LET DBA=%SCAN(&ARGLIST,2,%STR());
1  %LET RBA=%SCAN(&ARGLIST,3,%STR());
1  %LET DP=%SCAN(&ARGLIST,4,%STR());
1  %LET ACT=%SCAN(&ARGLIST,5,%STR());
4  %LET TTS=%SCAN(&ARGLIST,6,%STR());
4  %LET TEC=%SCAN(&ARGLIST,7,%STR());
4  %LET M_R=%SCAN(&ARGLIST,8,%STR());
4  %LET I_M=%SCAN(&ARGLIST,9,%STR());
4  %LET A_E=%SCAN(&ARGLIST,10,%STR());
4  %LET ADP=%SCAN(&ARGLIST,11,%STR());
4  %LET MCP=%SCAN(&ARGLIST,12,%STR());
4  %LET ENV=%SCAN(&ARGLIST,13,%STR());
4  %LET SSI=%SCAN(&ARGLIST,14,%STR());
4  %LET SEC=%SCAN(&ARGLIST,15,%STR());
4  %LET SUP=%SCAN(&ARGLIST,16,%STR());
4  %LET ARC=%SCAN(&ARGLIST,17,%STR());
4  %LET CHM=%SCAN(&ARGLIST,18,%STR());
4  %LET HYP=%SCAN(&ARGLIST,19,%STR());
4  %LET MUL=%SCAN(&ARGLIST,20,%STR());
4  %LET PLT=%SCAN(&ARGLIST,21,%STR());
4  %LET RGS=%SCAN(&ARGLIST,22,%STR());
4  %LET TRN=%SCAN(&ARGLIST,23,%STR());
4  %LET PRI_TYPE=%SCAN(&ARGLIST,24,%STR());

1  /*****
1  /* ARGUMENT LIST DEFINITIONS
8  START_FY, STARTING FY
24 SELECT FUNDING CATEGORIES
8 DBA, DBA
8 RBA, RBA
8 DP, DP
8 ACT, ACTIVITIES
4 FUNDING CATEGORIES ALL BLANK EQUAL SELECT ALL FUNDING CATEGORIES
24 SELECT PROGRAMS
8 TTS, TEST
8 TEC, TECHNOLOGY
8 M_R, MAINTENANCE AND REPAIR
8 I_M, IMPROVEMENT AND MODERNIZATION
8 A_E, ANALYSIS AND EVALUATION
8 ADP, AUTOMATIC DATA PROCESSING
8 MCP, MILITARY CONSTRUCTION PROGRAM
8 ENV, ENVIRONMENTAL
8 SSI, 373 PROGRAMS
8 SEC, SECURITY
8 SUP, SUPPLY
14 PROGRAMS ALL BLANK EQUAL SELECT ALL PROGRAMS
23 SELECT COMPLEXES
8 ARC, ARC'S
8 CHM, CHAMBERS
8 HYP, HYPERSONICS
8 MUL, MULTIPLE COMPLEXES
8 PLT, PLANTS
8 RGS, RANGES
8 TRN, TRANSONICS
14 COMPLEXES ALL BLANK EQUAL SELECT ALL COMPLEXES
21 PRIORITY TYPE
8 PRI_TYPE; X=TUIB PRIORITY
8 Y=SUBTASK PRIORITY
38 VY=INVESTMENT RRC PRIORITY
38 Z=CALSPAN INTEGRATED PRIORITY
38

1  */
1  /*****
1  /* DATA SET NAME CONTAINING DATA FOR REPORT
1  /*****
1  LIBNAME LRP 'CAL.SEUG.LRP' SERVER=SERVER8;

1  /*****
1  /* SET UP THE PRINTER PAGE
1  /*****
1  OPTIONS PS=60 LS=132 PAGENO=1;

1  /*****
1  /* IF NO STARTING FY SPECIFIED USE CURRENT FY AS DEFAULT
1  /*****
1  %IF (%NRBQUOTE(&START_FY) EQ %STR( )) %THEN %LET START_FY=%CUR_FY;

```

Fig. 9. PRJLIST MACRO SOURCE CODE.

```

4  /* USE INPUT PARAMETERS TO SELECT DATA FOR REPORT */
1  DATA SELECT;
3  SET LRP.LRP;
3  SELECT;
5  WHEN (
11 AFY GE "&START_FY") AND
11 (AFUND EQ 'Y' OR AFUND EQ 'N') AND
11 ((("&DBA" EQ AFC) OR
12 ("&RBA" EQ AFC) OR
12 ("&DP" EQ AFC) OR
12 ("&ACT" EQ AFC) OR
12 ("&DBA" EQ ' ' AND "&RBA" EQ ' ' AND
12 ("&DP" EQ ' ' AND "&ACT" EQ ' ')) AND
11 ((("&TTS" EQ APROG) OR
12 ("&TEC" EQ APROG) OR
12 ("&M_R" EQ APROG) OR
12 ("&I_M" EQ APROG) OR
12 ("&A_E" EQ APROG) OR
12 ("&ADP" EQ APROG) OR
12 ("&MCP" EQ APROG) OR
12 ("&ENV" EQ APROG) OR
12 ("&SSI" EQ APROG) OR
12 ("&SEC" EQ APROG) OR
12 ("&SUP" EQ APROG) OR
12 ("&TTS" EQ ' ' AND "&TEC" EQ ' ' AND "&M_R" EQ ' ' AND
13 "&I_M" EQ ' ' AND "&A_E" EQ ' ' AND "&ADP" EQ ' ' AND
13 "&MCP" EQ ' ' AND "&ENV" EQ ' ' AND "&SSI" EQ ' ' AND
13 "&SEC" EQ ' ' AND "&SUP" EQ ' ')) AND
11 ((("&ARC" EQ AMO) OR
12 ("&CHM" EQ AMO) OR
12 ("&HYP" EQ AMO) OR
12 ("&RGS" EQ AMO) OR
12 ("&TRN" EQ AMO) OR
12 ("&MUL" EQ AMO) OR
12 ("&ARC" EQ ' ' AND "&CHM" EQ ' ' AND "&HYP" EQ ' ' AND
13 "&RGS" EQ ' ' AND "&TRN" EQ ' ' AND "&MUL" EQ ' '))
9  );
7  OTHERWISE DELETE;
4  END;
4  PCMS=SUBSTR(APCMS,1,6);
1  RUN;

1  /*****
1  /* THE CODE ABOVE THIS POINT COULD BE MADE INTO A SEPARATE */
1  /* REUSABLE MACRO IF SEVERAL REPORTS USE THE SAME SELECT */
1  /* CRITERIA. */
1  /* THE CODE BELOW THIS POINT IS UNIQUE TO THIS REPORT */
1  /*****/

1  /*****
1  /* IF SELECTED PRIORITY IS MISSING SET IT TO A HIGH VALUE */
1  /* SO IT WILL SORT AT THE END OF THE LIST */
1  /*****/
2  DATA CHG_MISS;
5  SET SELECT;
5  IF (PRY&PRI_TYPE EQ .) THEN PRY&PRI_TYPE=9999;

1  /*****
1  /* SORT THE DATA BEFORE PRINTING */
1  /*****/
2  PROC SORT DATA=CHG_MISS;
5  BY PRY&PRI_TYPE;
1  /*****
1  /* CHANGE MISSING PRIORITIES BACK TO MISSING */
1  /*****/
2  DATA CHG_BACK;
5  SET CHG_MISS;
5  IF (PRY&PRI_TYPE EQ 9999) THEN PRY&PRI_TYPE=.;

1  /*****
1  /* PRINT THE REPORT */
1  /*****/
1  PROC PRINT NOOBS;
4  VAR PRYZ PRYYY EPR AMO APCMS ATITLE AFY APH
8  AST AFC AORG APRJENGR ADATE PTMHRM STMHRM TOTPRJC AFUND;
1  RUN;

1  /*****
1  /* RECALL THE MAIN MENU */
1  /*****/
1  %INCLUDE LIB(SEUGWIN) /SOURCE2;
1  %MEND PRJLIST;

```

Fig. 9. CONCLUDED.

```

1  /*****
1  /* COMPANY: CALSPAN/AEDC */
1  /* SOURCE CODE NAME: LRPMENU */
1  /* AUTHOR: ROBERT F. DAVIS */
1  /* PURPOSE: */
1  /* EXAMPLE CODE FOR USE AT SAS SOUTHEASTERN USER GROUP */
1  /* THIS CODE CREATES A MULTI PAGE MENU USING PROC PSEDIT AND */
1  /* PROCESSES SELECTIONS MADE BY A USER AND INCLUDES SOURCE TO */
1  /* CREATE A REPORT USING THE PROGRAM NAME PASSED AS THE ARGUMENT. */
1  /*****/

```

Fig. 10 LRPMENU MACRO SOURCE CODE.

```

1 %MACRO LRPMENU (PROGRAM);
1 %GLOBAL TITLE1;
1 %GLOBAL TITLE2;
1 %GLOBAL LASTPROG;
1 %GLOBAL PRI;
1 %LET ABORT= ' ';
1 DM 'PMENU OFF';

1 /*****
1 /*  ASSIGN DATA SET CONTAINING FSEDIT MENU SCREENS
1 /*****
1 LIBNAME LRP 'CAL.SEUG.LRP' SERVER=SERVER8;

1 RUN;
1 /*****
1 /*  DEFINE VARIABLES FOR FSEDIT MENU DATA
1 /*****
1 DATA LRPMENU;
4   ATTRIB PROGRAM LENGTH=$8 LABEL='CALLING*PROGRAM*NAME';
4   ATTRIB START_FY LENGTH=$4 LABEL='STARTING*YEAR';
4   ATTRIB END_FY LENGTH=$4 LABEL='ENDING*YEAR';
4   ATTRIB PLAN LENGTH=$1 LABEL='PLANNING';
4   ATTRIB DESIGN LENGTH=$1 LABEL='DESIGN';
4   ATTRIB EXEC LENGTH=$1 LABEL='EXECUTION';
4   ATTRIB YES LENGTH=$1 LABEL='FUND=YES';
4   ATTRIB NO LENGTH=$1 LABEL='FUND=NO';
4   ATTRIB BACK LENGTH=$1 LABEL='FUND=BACKLOG';
4   ATTRIB PRE LENGTH=$1 LABEL='PREBUYS';
4   ATTRIB READY LENGTH=$1 LABEL='READY*FOR*FUNDING';
4   ATTRIB DBA LENGTH=$1 LABEL='DBA';
4   ATTRIB RBA LENGTH=$1 LABEL='RBA';
4   ATTRIB DP LENGTH=$1 LABEL='DP';
4   ATTRIB ACT LENGTH=$1 LABEL='ACTIVITIES';
4   ATTRIB TTS LENGTH=$1 LABEL='TEST';
4   ATTRIB TEC LENGTH=$1 LABEL='TECHNOLOGY';
4   ATTRIB M_R LENGTH=$1 LABEL='M&R';
4   ATTRIB I_M LENGTH=$1 LABEL='I&M';
4   ATTRIB A_E LENGTH=$1 LABEL='A&E';
4   ATTRIB ADP LENGTH=$1 LABEL='ADP';
4   ATTRIB SEC LENGTH=$1 LABEL='SECURITY';
4   ATTRIB SUP LENGTH=$1 LABEL='SUPPLY';
4   ATTRIB ENV LENGTH=$1 LABEL='ENVIRONMENT';
4   ATTRIB MCP LENGTH=$1 LABEL='MCP';
4   ATTRIB SSI LENGTH=$1 LABEL='SSI';
4   ATTRIB ARC LENGTH=$1 LABEL='ARC';
4   ATTRIB CHM LENGTH=$1 LABEL='CHM';
4   ATTRIB HYP LENGTH=$1 LABEL='HYP';
4   ATTRIB RGS LENGTH=$1 LABEL='RGS';
4   ATTRIB TRN LENGTH=$1 LABEL='TRN';
4   ATTRIB MUL LENGTH=$1 LABEL='MULTIPLE*COMPLEXES';
4   ATTRIB MPRYX LENGTH=$1 LABEL='MISSING*PRIORITY*X';
4   ATTRIB MPRYY LENGTH=$1 LABEL='MISSING*PRIORITY*Y';
4   ATTRIB MPRYY LENGTH=$1 LABEL='MISSING*PRIORITY*YY';
4   ATTRIB MPRYZ LENGTH=$1 LABEL='MISSING*PRIORITY*Z';
4   ATTRIB OPRYX LENGTH=$1 LABEL='OUTPUT*PRIORITY*X';
4   ATTRIB OPRYY LENGTH=$1 LABEL='OUTPUT*PRIORITY*Y';
4   ATTRIB OPRYY LENGTH=$1 LABEL='OUTPUT*PRIORITY*YY';
4   ATTRIB OPRYZ LENGTH=$1 LABEL='OUTPUT*PRIORITY*Z';
4   ATTRIB SPRYX LENGTH=2 LABEL='SORT*ON*PRIORITY*X';
4   ATTRIB SPRYY LENGTH=2 LABEL='SORT*ON*PRIORITY*Y';
4   ATTRIB SPRYY LENGTH=2 LABEL='SORT*ON*PRIORITY*YY';
4   ATTRIB SPRYZ LENGTH=2 LABEL='SORT*ON*PRIORITY*Z';
4   ATTRIB SPFY LENGTH=2 LABEL='SORT*ON*FY';
4   ATTRIB SPC LENGTH=2 LABEL='SORT*ON*FUNDING*CATEGORY';
4   ATTRIB SPH LENGTH=2 LABEL='SORT*ON*PHASE';
4   ATTRIB SFUND LENGTH=2 LABEL='SORT*ON*FUND*CODE';
4   ATTRIB SORG LENGTH=2 LABEL='SORT*ON*ORG*CODE';
4   ATTRIB SPCMS LENGTH=2 LABEL='SORT*ON*PCMS#';
4   ATTRIB SPROG LENGTH=2 LABEL='SORT*ON*PROGRAM';
4   ATTRIB SENGLAST LENGTH=2 LABEL='SORT*ON*ENGINEERS*LAST*NAME';
4   ATTRIB SJOB LENGTH=2 LABEL='SORT*ON*JOB#';
4   ATTRIB SAMO LENGTH=2 LABEL='SORT ON COMPLEX';
4   ATTRIB TITLE1 LENGTH=$75 LABEL='REPORT TITLE LINE 1';
4   ATTRIB TITLE2 LENGTH=$75 LABEL='REPORT TITLE LINE 2';

1 /*****
1 /*  IF WE HAVE ALREADY RUN AND ARE REQUESTING THE SAME MENU OPTION
1 /*  KEEP THE PREVIOUS SETTINGS BY USING THE SET STATEMENT.
1 /*****
1 %IF (%NRQUOTE(&PROGRAM) EQ %NRQUOTE(&LASTPROG)) %THEN %DO;
4   SET LRPMENU;
1 %END;
1 RUN;

1 /*****
1 /*  BEGIN A MULTI PAGE MENU
1 /*****
1 PROC FSEDIT DATA=LRPMENU SCREEN=SEUG.LRPMENU.LRPMENU;
1 RUN;

1 /*****
1 /*  IF USER DID NOT PRESS ABORT PF KEY THEN PROCESS MENU DATA
1 /*****
1 %IF (%NRQUOTE(&ABORT) NE %STR(X)) %THEN %DO;

```

Fig. 10. CONTINUED.

```

1 DATA SELECT;
2 SET LRP.LRP(CNTLLEV=REC);
3 IF (_N_=1) THEN
4 SET LRPMENU;
5 SELECT;
6 WHEN (
11 ((START_FY EQ ' ' AND AFY GE &CUR_FY) OR
12 (END_FY EQ ' ' AND AFY GE START_FY) OR
12 (AFY GE START_FY AND AFY LE END_FY)) AND
11 ((PLAN EQ 'X' AND APH EQ 'P') OR
12 (DESIGN EQ 'X' AND APH EQ 'D') OR
12 (EXEC EQ 'X' AND
15 (APH EQ 'NP' OR APH EQ 'MY' OR APH EQ 'CO'))
38 OR
12 ((PLAN EQ ' ' AND DESIGN EQ ' ' AND EXEC EQ ' ')) AND
11 ((YES EQ 'X' AND AFUND EQ 'Y') OR
12 (NO EQ 'X' AND AFUND EQ 'N') OR
12 (BACK EQ 'X' AND AFUND EQ 'B') OR
12 (PRE EQ 'X' AND AFUND EQ 'P') OR
12 (READY EQ 'X' AND AFUND EQ 'R') OR
12 (YES EQ ' ' AND NO EQ ' ' AND BACK EQ ' ' AND
13 PRE EQ ' ' AND READY EQ ' ')) AND
11 ((DBA EQ 'X' AND AFC EQ 'DBA') OR
12 (RBA EQ 'X' AND AFC EQ 'RBA') OR
12 (DP EQ 'X' AND AFC EQ 'DP') OR
12 (ACT EQ 'X' AND AFC EQ 'ACT') OR
12 (DBA EQ ' ' AND RBA EQ ' ' AND
13 DP EQ ' ' AND ACT EQ ' ')) AND
11 ((TTS EQ 'X' AND APROG EQ 'TTS') OR
12 (TEC EQ 'X' AND APROG EQ 'TEC') OR
12 (M_R EQ 'X' AND APROG EQ 'M&R') OR
12 (I_M EQ 'X' AND APROG EQ 'I&M') OR
12 (A_E EQ 'X' AND APROG EQ 'A&E') OR
12 (ADP EQ 'X' AND APROG EQ 'ADP') OR
12 (MCP EQ 'X' AND APROG EQ 'MCP') OR
12 (ENV EQ 'X' AND APROG EQ 'ENV') OR
12 (SSI EQ 'X' AND APROG EQ 'SSI') OR
12 (SEC EQ 'X' AND APROG EQ 'SEC') OR
12 (SUP EQ 'X' AND APROG EQ 'SEC') OR
12 (TTS EQ ' ' AND TEC EQ ' ' AND M_R EQ ' ' AND
13 I_M EQ ' ' AND A_E EQ ' ' AND ADP EQ ' ' AND
13 MCP EQ ' ' AND ENV EQ ' ' AND SSI EQ ' ' AND
13 SEC EQ ' ' AND SUP EQ ' ')) AND
11 ((ARC EQ 'X' AND AMO EQ 'ARC') OR
12 (CHM EQ 'X' AND AMO EQ 'CHM') OR
12 (HYP EQ 'X' AND AMO EQ 'HYP') OR
12 (RGS EQ 'X' AND AMO EQ 'RGS') OR
12 (TRN EQ 'X' AND AMO EQ 'TRN') OR
12 (MUL EQ 'X' AND AMO EQ 'MUL') OR
12 (ARC EQ ' ' AND CHM EQ ' ' AND HYP EQ ' ' AND
13 RGS EQ ' ' AND TRN EQ ' ' AND MUL EQ ' ')) AND
11 ((MPRYX EQ 'X' AND PRYX EQ ' ') OR
12 (MPRYX EQ 'X' AND PRYX EQ ' ') OR
12 (MPRYXX EQ 'X' AND PRYXX EQ ' ') OR
12 (MPRYZ EQ 'X' AND PRYZ EQ ' ') OR
12 (MPRYX EQ ' ' AND MPRYY EQ ' ' AND
12 (MPRYXX EQ ' ' AND MPRYZ EQ ' '))
9 );
7 OTHERWISE DELETE;
4 END;
4 PCMS=SUBSTR(APCMS,1,6);

1 /*****
1 /* STORE USER TITLES IN GLOBAL COMMON */
1 /*****
4 IF (TITLE1 NE ' ') THEN
7 CALL SYMPUT('TITLE1',TRIM(TITLE1));
4 ELSE CALL SYMPUT('TITLE1',' ');
4 IF (TITLE2 NE ' ') THEN CALL SYMPUT('TITLE2',TRIM(TITLE2));
4 ELSE CALL SYMPUT('TITLE2',' ');

4 /* SET PRIORITY TYPE TO BE OUTPUT ON REPORT */
4 IF (OPRYX EQ 'X') THEN CALL SYMPUT('PRI','X');
4 ELSE IF (OPRYX EQ 'X') THEN CALL SYMPUT('PRI','Y');
4 ELSE IF (OPRYXX EQ 'X') THEN CALL SYMPUT('PRI','YY');
4 ELSE CALL SYMPUT('PRI','Z');
1 RUN;

1 /*****
1 /* CREATE TITLES FOR REPORT */
1 /*****
1 %LRPTITLE;
1 /*****
1 /* PROCESS SORT LIST FROM MENU SCREEN 2 */
1 /*****
1 DATA SRTPARAM(KEEP=NAME NUM);
1 ARRAY NAM (*) $8 NAMI-NAM13 ('PRYX','PRYY','PRYYY','PRYZ','AFY',
18 'AFUND','AFC','APH','PCMS','AORG','AENGLAST',
18 'APROG','AJOB','AMO');
1 ARRAY CNT (*) SPRYX SPRYY SPRYYY SPRYZ SFY SFUND SFC SPH SPCMS
15 SORG SENGLAST SPROG SJOB SAMO;
1 SET LRPMENU;
1 DO I=1 TO DIM(CNT);
1 NAME = NAM{I};
1 NUM = CNT{I};

```

Fig. 10. CONTINUED.

```

1 OUTPUT;
1 END;

1 DATA SRTPARAM;
1 SET SRTPARAM;
1 IF NUM EQ . THEN NUM = 999;

1 PROC SORT DATA=SRTPARAM;
1 BY NUM;

1 DATA SRTPARAM;
1 SET SRTPARAM;
1 IF NUM EQ 999 THEN NUM = .;

1 %MACRO SORT_VAR;

1 DATA Y;
1 SET SRTPARAM;
1 IF (NUM EQ .) THEN CALL SYMPUT('VAR' || LEFT(_N_), ' ');
1 ELSE CALL SYMPUT('VAR' || LEFT(_N_), NAME);
1 CALL SYMPUT('BY',
3 'BY &VAR1 &VAR2 &VAR3 &VAR4 &VAR5 &VAR6 &VAR7 &VAR8 &VAR9 &VAR10');
1 RUN;
1 %MEND;
1 %SORT_VAR;
1 %INCLUDE LIB(&PROGRAM) /SOURCE2;
1 %END;

1 /*****
1 /* IF USER PRESSED ABORT PF KEY THEN RECALL MAIN MENU */
1 /*****
1 %ELSE %DO;
7 %INCLUDE LIB(SEUGWIN) /SOURCE2;
1 %END;
1 RUN;

1 %MEND LRPMENU;

```

02130006 *
02140006 *
02150006 *
02160006 *
02170006 *
02180006 *
02190006 *
02200006 *
02210006 *
02220006 *
02230006 *
02240006 *
02250006 *
02260006 *
02270006 *
02280006 *
02290006 *
02300006 *
02310006 *
02320006 *
02330006 *
02340006 *
02350006 *
02360006 *
02370006 *
02380006 *
02390006 *
02400009 *
02410009 *
02420009 *
02430009 *
02440006 *
02450006 *
02460006 *
02470006 *
02480006 *
02490006 *

Fig. 10. CONCLUDED.

FSEDIT WORK.LRPMENU	Obs 1 Screen 1
Command ==>	
NOTE: This application uses 2 screens.	
LONG RANGE PLANNING REPORT SELECTION CRITERIA FOR REPORT STFRM	
FILL IN RANGE OF FY'S: STARTING FY: 1994 ENDING FY: ____	
MARK ITEM(S) TO BE USED AS SELECT CRITERIA WITH AN X:	
PHASE(S): PLANNING (P): _ DESIGN (D): _ EXECUTION (NP,MY,CO): _	
FUNDING: Y(ES): _ N(O): _ B(ACKLOG): _ P(REBUY): _ R(EADY FOR FUNDING): _	
FUNDING CATEGORY: DBA: _ RBA: _ DP: _ ACT: _	
PROGRAMS: TTS: _ TEC: _ M&R: _ I&M: _ A&E: _ ADP: _ MCP: _ ENV: _	
SSI: _ SEC: _ SUP: _	
COMPLEX: ARC: _ CHM: _ HYP: _ RGS: _ TRN: _ MUL: _	
SELECT ONLY RECORDS WITH MISSING PRIORITIES:	
TUIB(PRYX) _ SUBTASK(PRY) _ INVESTMENT RRC(PRYYY) _ INTEG(PRYZ) _	
SELECT PRIORITY TO PRINT ON REPORT BY MARKING ONLY ONE WITH AN X.	
TUIB(PRYX) _ SUBTASK(PRY) _ INVESTMENT RRC(PRYYY) _ INTEG(PRYZ) _	
PF8= SORT SELECTION SCREEN. PF3= END SELECTION AND EXECUTE PF5= ABORT	

Fig. 11. MULTI PAGE MENU SCREEN 1 WINDOW.

FSEDIT WORK.LRPMENU	Obs 1 Screen 2
Command ==>	
OUTPUT SORT CRITERIA	
ENTER NUMBERS 1 THRU X FOR SORT SEQUENCE:	
TUIB PRIORITY (PRYX): _	PCMS # (1ST 6 DIGITS): _
SUBTASK PRIORITY (PRY): _	PHASE (APH): _
INVESTMENT RRC PRIORITY (PRYYY): _	FY (AFY): _
CALSPAN INTEGRATED PRIORITY (PRYZ): _	FUNDING (AFUND): _
ORGANIZATION (AORG): _	FUNDING CATEGORY (AFC): _
ENGINEER'S LASTNAME (AENGLAST): _	PROGRAM (APROG): _
AIMS JOB # (AJOB): _	COMPLEX (AMO): _
DEFAULT TITLE: CALSPAN LONG RANGE PLANNING DATA FOR:	
OPTIONAL TITLE(S) OVERRIDE DEFAULT	
OPTIONAL TITLE LINE 1:	
OPTIONAL TITLE LINE 2:	
PF7= SELECT CRITERIA SCREEN PF3= END SELECTIONS AND EXECUTE PF5= ABORT	

Fig. 12. MULTI PAGE MENU SCREEN 2 WINDOW.

```

1  /*****
1  /* COMPANY: CALSPAN/AEDC
1  /* SOURCE CODE NAME: MENUSCL
1  /* AUTHOR: ROBERT F. DAVIS
1  /* PURPOSE:
1  /* EXAMPLE CODE FOR USE AT SAS SOUTHEASTERN USER GROUP.
1  /* THIS IS SCREEN CONTROL LANGUAGE USED IN AN FSEDIT SESSION OF
1  /* THE CALSPAN PLANNING AND WORK CONTROL CENTER MENU SYSTEM.
1  /*****
1  ARRAY SORTCHK {*} CHK1-CHK13;
1  LENGTH LASTPROG $ 8;
1  FSEINIT:
1  PROTECT PROGRAM; /* PROTECT PROGRAM FIELD ON MENU */
1  CALL SYMPUT('ABORT',' '); /* CLEAR THE ABORT FLAG */
1  CONTROL ENTER ALWAYS;
1  RETURN;

1  INIT:
1  CURSOR START_FY; /* POSITION CURSOR TO START FY FIELD */
1  PROGRAM=SYMGET('PROGRAM'); /* GET CURRENT PROGRAM NAME */
1  LASTPROG=SYMGET('LASTPROG'); /* AND LAST PROGRAM NAME FROM GLOBALS */
1  IF (LASTPROG NE PROGRAM) THEN /* IF NO PREVIOUS PROGRAM */
4  START_FY=SYMGETN('CUR_FY'); /* USE CURRENT FY AS START FY */
1  RETURN;

1  MAIN:
1  CVAL=WORD(1,'U');
1  IF (CVAL EQ 'ABORT') THEN DO; /* IF USER PRESSED ABORT PF KEY */
4  CALL SYMPUT('ABORT','X'); /* SET ABORT FLAG */
4  CALL EXECMD('END'); /* AND EXIT */
1  END;
1  RETURN;

1  TERM:
1  CVAL=SYMGET('ABORT');
1  /*****
1  /* VERIFY AT LEAST ONE SORT PARAMETER WAS SPECIFIED
1  /*****
1  IF (CVAL NE 'X') THEN DO;
4  IF (SPRYX LT 1 AND SPRYY LT 1 AND SPRYYY LT 1 AND
8  SPRYZ LT 1 AND SPCMS LT 1 AND SPH LT 1 AND
8  SFY LT 1 AND SFUND LT 1 AND SFC LT 1 AND
8  SJOB LT 1 AND SPROG LT 1 AND SORG LT 1 AND
8  SENGLAST LT 1) THEN DO;
8  CURSOR SPRYX;
7  _STATUS_='R';
7  _MSG_='AT LEAST ONE SORT PARAMETER MUST BE DEFINED';
4  END;
1  /*****
1  /* VERIFY ONLY ONE OUTPUT PRIORITY WAS SELECTED
1  /*****
1  IF ((OPRYX EQ 'X' AND
9  (OPRYY EQ 'X' OR OPRYYY EQ 'X' OR OPRYZ EQ 'X')) OR
8  (OPRYY EQ 'X' AND
9  (OPRYX EQ 'X' OR OPRYYY EQ 'X' OR OPRYZ EQ 'X')) OR
8  (OPRYYY EQ 'X' AND
9  (OPRYX EQ 'X' OR OPRYY EQ 'X' OR OPRYZ EQ 'X')) OR
9  (OPRYZ EQ 'X' AND
8  (OPRYX EQ 'X' OR OPRYY EQ 'X' OR OPRYYY EQ 'X'))
7  ) THEN DO;
7  CURSOR OPRYX;
7  _STATUS_='R';
7  _MSG_='ONLY ONE OUTPUT PRIORITY ALLOWED.';
4  END;
1  /*****
1  /* VERIFY SORT NUMBERS WHERE ONLY USED ONCE
1  /* NO REPEAT OF SORT NUMBERS ALLOWED
1  /*****
4  DO J=1 TO 13;
7  SORTCHK{J}=0;
4  END;
4  IF (SPRYX NE .) THEN SORTCHK{SPRYX}+1;
4  IF (SPRYY NE .) THEN SORTCHK{SPRYY}+1;
4  IF (SPRYYY NE .) THEN SORTCHK{SPRYYY}+1;
4  IF (SPRYZ NE .) THEN SORTCHK{SPRYZ}+1;
4  IF (SPCMS NE .) THEN SORTCHK{SPCMS}+1;
4  IF (SPH NE .) THEN SORTCHK{SPH}+1;
4  IF (SFY NE .) THEN SORTCHK{SFY}+1;
4  IF (SFUND NE .) THEN SORTCHK{SFUND}+1;
4  IF (SFC NE .) THEN SORTCHK{SFC}+1;
4  IF (SJOB NE .) THEN SORTCHK{SJOB}+1;
4  IF (SPROG NE .) THEN SORTCHK{SPROG}+1;
4  IF (SORG NE .) THEN SORTCHK{SORG}+1;
4  IF (SENGLAST NE .) THEN SORTCHK{SENGLAST}+1;
4  DO J=1 TO 13;
7  IF (SORTCHK{J} GT 1) THEN DO;
10  _MSG_='PLEASE DO NOT DUPLICATE SORT NUMBERS';
10  _STATUS_='R';
7  END;
4  END;
1  END;
1  RETURN;

1  FSETERM:
1  /*****

```

Fig. 13. FSEDIT SCREEN CONTROL LANGUAGE SOURCE CODE.

```

1  /*      STORE THIS PROGRAM NAME FOR NEXT TIME      */ 01070009 *
1  /****** */ 01080009 *
1  CALL SYMPUT('LASTPROG',PROGRAM); 01090009 *
1  RETURN; 01100009 *

```

Fig. 13. CONCLUDED.

```

1  /****** */ 00030007 *
1  /* COMPANY: CALSPAN/AEDC */ 00050007 *
1  /* SOURCE CODE NAME: LRPTITLE */ 00070009 *
1  /* AUTHOR: ROBERT F. DAVIS */ 00090007 *
1  /* PURPOSE: */ 00110007 *
1  /* EXAMPLE CODE FOR USE AT SAS SOUTHEASTERN USER GROUP. */ 00120015 **
1  /* THIS CODE FORMATS TITLES SUPPLIED BY A USER AND PASSED */ 00130015 **
1  /* BY PROGRAM LRPMENU. */ 00140015 **
1  /****** */ 00180007 *
1  %MACRO LRPTITLE; 00200005 *
1  DATA TITLES; 00220005 *
1  ATTRIB LINE1 LENGTH=$132; 00230005 *
1  ATTRIB LINE2 LENGTH=$132; 00240005 *
1  /****** */ 00250012 *
1  /* DEFAULT TITLE TO USE IF USER DID NOT SUPPLY ONE */ 00260012 *
1  /****** */ 00270012 *
1  RETAIN ALTTITLE 'CALSPAN LONG RANGE PLANNING DATA FOR: '; 00280005 *
1  %DO I=1 %TO 2; 00290005 *
1  /****** */ 00300012 *
1  /* CENTER TITLE 1 */ 00310012 *
1  /****** */ 00320012 *
1  %IF (&I EQ 1) %THEN %DO; 00330005 *
1  %IF (%NRQUOTE("&&TITLE&I") NE %STR(" ")) %THEN %DO; 00340005 *
1  L= ((132-%LENGTH("&&TITLE&I"))/2)-%LENGTH(&PROGRAM)-1; 00350005 *
10  LINE1= "&PROGRAM" || REPEAT(' ',L) || %NRQUOTE("&&TITLE&I"); 00360005 *
10  CALL SYMPUT('LINE1',LINE1); 00370005 *
1  %END; 00380005 *
1  %ELSE %DO; 00390005 *
10  L= ((132-%LENGTH(ALTTITLE))/2)-%LENGTH(&PROGRAM)-1; 00400005 *
10  LINE1= "&PROGRAM" || REPEAT(' ',L) || ALTTITLE; 00410005 *
10  CALL SYMPUT('I',2); 00420005 *
10  CALL SYMPUT('LINE1',LINE1); 00430005 *
10  %GOTO PARAMS; 00440005 *
1  %END; 00450005 *
1  %END; 00460005 *
1  %ELSE %DO; 00470005 *
1  /****** */ 00480012 *
1  /* CENTER TITLE 2 */ 00490012 *
1  /****** */ 00500012 *
1  %IF (%NRQUOTE("&&TITLE&I") NE %STR(" ")) %THEN %DO; 00510005 *
10  L= ((132-%LENGTH("&&TITLE&I"))/2)-1; 00520005 *
10  LINE2= REPEAT(' ',L) || %NRQUOTE("&&TITLE&I"); 00530005 *
10  CALL SYMPUT('LINE2',LINE2); 00540005 *
1  %END; 00550005 *
1  %ELSE %DO; 00560005 *
10  %GOTO PARAMS; 00570005 *
1  %END; 00580005 *
1  %END; 00590005 *
1  %END; 00600005 *
1  %PARAMS; 00610005 *
1  RUN; 00620005 *
1  /****** */ 00630005 *
1  /* CREATE THIRD TITLE WITH USER SELECT CRITERIA */ 00640012 *
1  /****** */ 00650012 *
1  DATA PARAMS; 00660012 *
1  ATTRIB PARAMS LENGTH=$200; 00670005 *
1  ATTRIB PARAM1 LENGTH=$132; 00680005 *
1  ATTRIB PARAM2 LENGTH=$132; 00690005 *
1  SET LRPMENU; 00700005 *
10  IF (START_FY NE ' ') THEN PARAMS= 'FY' || START_FY; 00710005 *
10  ELSE PARAMS='FY' || &CUR_FY; 00720005 *
10  IF (END_FY NE ' ') AND (START_FY NE END_FY) 00730005 *
14  THEN PARAMS=TRIM(PARAMS) || '-' || END_FY || ' '; 00740005 *
10  ELSE IF ((END_FY EQ ' ') OR (END_FY EQ '-')) 00750005 *
15  THEN PARAMS=TRIM(PARAMS) || '+'; 00760005 *
10  IF (PLAN EQ 'X' OR DESIGN EQ 'X' OR EXEC EQ 'X') THEN DO; 00770005 *
13  PARAMS=TRIM(PARAMS) || ' PHASE(S)='; 00780005 *
13  IF (PLAN EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' PLANNING'; 00790005 *
13  IF (DESIGN EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' DESIGN'; 00800005 *
13  IF (EXEC EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' EXECUTION'; 00810005 *
13  PARAMS=TRIM(PARAMS) || ' '; 00820005 *
10  END; 00830005 *
10  IF (YES EQ 'X' OR NO EQ 'X' OR BACK EQ 'X' OR 00840005 *
14  PRE EQ 'X' OR READY EQ 'X') THEN DO; 00850005 *
13  PARAMS=TRIM(PARAMS) || ' FUNDED='; 00860005 *
13  IF (YES EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' Y'; 00870005 *
13  IF (NO EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' N'; 00880005 *
13  IF (BACK EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' B'; 00890005 *
13  IF (PRE EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' P'; 00900005 *
13  IF (READY EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' R'; 00910005 *
13  PARAMS=TRIM(PARAMS) || ' '; 00920005 *
10  END; 00930005 *
10  IF (DBA EQ 'X' OR RBA EQ 'X' OR DP EQ 'X' OR ACT EQ 'X') 00940005 *
11  THEN DO; 00950005 *
13  PARAMS=TRIM(PARAMS) || ' FUND CATEGORY='; 00960005 *
13  IF (DBA EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' DBA'; 00970005 *
13  IF (RBA EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' RBA'; 00980005 *
13  00990005 *

```

Fig. 14. LRPTITLE MACRO SOURCE CODE

```

13 IF (DP EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' DP';
13 IF (ACT EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' ACT';
13 PARAMS=TRIM(PARAMS) || ' ';
10 END;
10 IF (TTS EQ 'X' OR TEC EQ 'X' OR M_R EQ 'X' OR I_M EQ 'X' OR
14 A_E EQ 'X' OR ADP EQ 'X' OR SEC EQ 'X' OR SUP EQ 'X' OR
14 ENV EQ 'X' OR MCP EQ 'X' OR SSI EQ 'X') THEN DO;
13 PARAMS=TRIM(PARAMS) || ' PROGRAM(S)=';
13 IF (TTS EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' TEST';
13 IF (TEC EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' TECHNOLOGY';
13 IF (M_R EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' M&R';
13 IF (I_M EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' I&M';
13 IF (A_E EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' A&E';
13 IF (ADP EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' ADP';
13 IF (SEC EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' SECURITY';
13 IF (SUP EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' SUPPLY';
13 IF (ENV EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' ENVIRONMENT';
13 IF (MCP EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' MCP';
13 IF (SSI EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' SSI';
13 PARAMS=TRIM(PARAMS) || ' ';
10 END;
10 IF (ARC EQ 'X' OR CHM EQ 'X' OR HYP EQ 'X' OR
14 RGS EQ 'X' OR TRN EQ 'X' OR MUL EQ 'X') THEN DO;
13 PARAMS=TRIM(PARAMS) || ' COMPLEX(ES)=';
13 IF (ARC EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' ARC';
13 IF (CHM EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' CHM';
13 IF (HYP EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' HYP';
13 IF (RGS EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' RGS';
13 IF (TRN EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' TRN';
13 IF (MUL EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' MUL';
13 PARAMS=TRIM(PARAMS) || ' ';
10 END;
10 IF (MPRYX EQ 'X' OR MPRYY EQ 'X' OR MPRYYY EQ 'X' OR
14 MPRYZ EQ 'X') THEN DO;
13 PARAMS=TRIM(PARAMS) || ' MISSING PRIORITY';
13 IF (MPRYX EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' X';
13 IF (MPRYY EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' Y';
13 IF (MPRYYY EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' YY';
13 IF (MPRYZ EQ 'X') THEN PARAMS=TRIM(PARAMS) || ' Z';
13 PARAMS=TRIM(PARAMS) || ' ';
10 END;
10 IF (LENGTH(PARAMS) GT 132) THEN DO;
13 POS=INDEX(REVERSE(SUBSTR(PARAMS,1,132)), ' ');
13 POS = 132-POS;
13 PARAM1=SUBSTR(PARAMS,1,POS);
13 PARAM2=SUBSTR(PARAMS,POS+1,200-POS);
13 CALL SYMPUT('PARAM1',PARAM1);
13 CALL SYMPUT('PARAM2',PARAM2);
13 CALL SYMPUT('CNT',2);
10 END;
10 ELSE DO;
12 CALL SYMPUT('PARAM1',PARAMS);
12 CALL SYMPUT('CNT',1);
10 END;
10 RUN;
1
1 /*****
1 /* CREATE SAS TITLE STATEMENTS FOR REPORT */
1 /*****
1 TITLE1 %NRQUOTE("&LINE1");
1 %IF (&I EQ 3) %THEN %DO;
4 TITLE2 %NRQUOTE("&LINE2");
1 %END;
1 %IF (&CNT EQ 2) %THEN %DO;
4 TITLE&I %NRQUOTE("&PARAM1");
4 TITLE%&VAL(&I+1) %NRQUOTE("&PARAM2");
1 %END;
1 %ELSE
4 TITLE&I %NRQUOTE("&PARAM1");
1 RUN;
1 %MEND LRPTITLE;

```

Fig. 14. CONCLUDED.

```

1 /***** 00030004 *
1 /* COMPANY: CALSPAN/AEDC */ 00050004 *
1 /* SOURCE CODE NAME: STFRM */ 00070004 *
1 /* AUTHOR: ROBERT P. DAVIS */ 00090004 *
1 /* PURPOSE: */ 00110004 *
1 /* EXAMPLE CODE FOR USE AT SAS SOUTHEASTERN USER GROUP. */ 00120007 **
1 /* THIS PROGRAM IS CALLED BY THE CALSPAN PLANNING AND WORK */ 00130004 *
1 /* CONTROL MENU SYSTEM FROM THE MULTI PAGE REPORT OPTION TO */ 00140007 **
1 /* OUTPUT A REPORT IN THE STANDARD SUBTASK FORMAT. */ 00150007 **
1 /***** 00180004 *
1 OPTIONS PS=60 LS=132 PAGENO=1; 00190004 *
1 00200004 *
2 PROC SORT DATA=SELECT; 00210004 *
1 /***** 00220004 *
1 /* SORT ON THE CREATED BY STATEMENT IN GLOBAL COMMON */ 00230004 *
1 /***** 00240004 *
3 &BY; 00250004 *
1 00260004 *
1 /***** 00270004 *
1 /* CREATE AND PRINT THE REPORT */ 00280004 *

```

Fig. 15. STFRM SOURCE CODE.

```

1  /*****
1  DATA NULL;
6  LENGTH TTITLE $30;
6  RETAIN TTOTGFE 0 TTOTMTL 0 TOCON 0 TPTMHRS 0 TPTMHRC 0
11 TSTMHRS 0 TSTMHRC 0 TTOTPRJC 0;
6  SET SELECT END=EOF;
6  FILE PRINT HEADER=HDR PS=60 N=PS LS=132;
6  TTITLE=TRIM(ATITLE);
10 TTOTGFE+TOTGFE;
10 TTOTMTL+TOTMTL;
10 TOCON+OCON;
10 TPTMHRS+PTMHRS;
10 TPTMHRC+PTMHRC;
10 TSTMHRS+STMHRS;
10 TSTMHRC+STMHRC;
10 TTOTPRJC+TOTPRJC;
6  PUT @ 1 APCMS @ 10 PRY&PRI 3.
10 @ 14 AJOB @ 20 TTITLE
10 @ 51 AFUND @ 55 AORG
10 @ 58 AFY @ 64 APH
10 @ 66 TOTGFE 8. @ 74 OCON 7.
10 @ 82 TOTMTL 7. @ 89 PTMHRS 7.
10 @ 98 PTMHRC 7. @ 107 STMHRS 6.
10 @ 115 STMHRC 7. @ 123 TOTPRJC 8.;
6  PUT @ 66 TTOTGFE 8. @ 74 TOCON 7.
10 @ 82 TTOTMTL 7. @ 89 TPTMHRS 7.
10 @ 98 TPTMHRC 7. @ 107 TSTMHRS 6.
10 @ 115 TSTMHRC 7. @ 123 TTOTPRJC 8.;
4  IF (EOF) THEN
6  PUT /// @ 30 'END OF DATA';
5  RETURN;
1  HDR:
13 PUT @ 1 'CECORS' @ 9 'PRY' '&PRI'
17 @ 15 'AIMS' @ 72 'M A T E R I A L'
17 @ 91 'FACILITY/DEPT' @ 110 'SUPPORT'
17 @ 125 'TOTAL';
13 PUT @ 1 'NO.' @ 14 'JOB NO.'
17 @ 21 'TITLE' @ 50 'FUND'
17 @ 55 'ORG' @ 59 'FY'
17 @ 62 'PHASE' @ 70 'GFE'
17 @ 76 'ONCON' @ 82 'OFFCON'
17 @ 90 'M/H' @ 100 'COST'
17 @ 107 'M/H' @ 115 'COST'
17 @ 125 'COST';
4  PUT @1 132*'-';
6  RETURN;
1  RUN;
1  /*****
1  /* RECALL THE MAIN MENU */
1  /*****
1  %INCLUDE LIB(SEUGWIN) /SOURCE2;

```

Fig. 15. CONCLUDED.

SEUGWIN	
REPORTS PRINT ... EXIT	
SINGLE PAGE MENU...	CALSPAN WORK CONTROL MAIN MENU
MULTI PAGE MENU	
TSO BATCH JOB...	TSO BATCH JOB...
CANCEL	IF SELECTED A BATCH JOB WILL BE SUBMITTED
EXIT	FOR RUN AFTER 1600 HOURS.
	. RUN MASTER
USE TAB KEY TO MA	[OK] [Cancel]
AND PRESS ENTER.	
F5=VIEW OUTPUT. F6=RETURN FROM VIEW OUTPUT. F7=PAGE BACKWARD. F8=PAGE FORWARD F10=PAGE LEFT. F11=PAGE RIGHT.	

Fig. 16 TSO BATCH JOB WINDOW

```

1 //B30518A JOB (CAL,RUNSASAA,01,51017655) 00010000
1 // 'B30518.ROBERT',PRTY=7,MSGLEVEL=(1,1),CLASS=X, 00020024
1 // NOTIFY=B30518,TIME=4,MSGCLASS=O,TYPRUN=HOLD 00030067
1 /*JOBPARM ROOM=17A 00040044
1 // EXEC SAS6 00050065
1 //SYSIN DD DATA,DLM=ZQ 00060000
1 00070069
1 /***** 00080069
1 /* COMPANY: CALSPAN/AEDC */ 00090069
1 /* SOURCE CODE NAME: TFRBAT */ 00100069
1 /* AUTHOR: ROBERT F. DAVIS */ 00140069

```

Fig. 17. TSO BATCH JOB SOURCE CODE.

```

1  /* PURPOSE:                                     */ 00160069 *
1  /*   EXAMPLE BATCH JOB TO BE SUBMITTED BY THE CALSPAN PLANNING   */ 00170069 *
1  /*   AND WORK CONTROL MENU SYSTEM.  FOR USE AT SAS SOUTHEASTERN   */ 00180069 *
1  /*   USERS GROUP.                                                */ 00190069 *
1  /******                                                    */ 00220069 *
1  FILENAME LIB 'CAL.SEUG.LIB';                               00230064 *
9  OPTIONS SOURCE MACROGEN MRECALL MAUTOSOURCE SASAUTOS=LIB      00240055 *
1  SYMBOLGEN MLOGIC;                                           00250064 *
1  %TPRMIRM(PROGRAM=TEC, PHASE=E, FUND_CAT=DBA, FY=1995, PRI=Z);  00260066 *
1  %TPRMIRM(PROGRAM=TTS, PHASE=E, FUND_CAT=DBA, FY=1995, PRI=Z);  00270066 *
1  ZQ                                                         00280066 *

```

Fig. 17. CONCLUDED.

SEUGWIN	PRI-PRINT
REPORTS	ENTER REMOTE PRINTER DESTINATION
	DEFAULT = RMT107
REPORT	ENTER NUMBER OF COPIES (1-99)
PRINT	DEFAULT = 1
EXIT -	
USE TAB K AND PRESS	PLEASE MAKE SELECTIONS USING THE TAB AND ENTER KEY. WHEN ALL SELECTIONS HAVE BEEN MADE, SELECT OK TO CONTINUE OR CANCEL TO ABORT.
	[OK] [Cancel]
F5=VIEW OUTPUT.	F6=RETURN FROM VIEW OUTPUT. F7=PAGE BACKWARD.
F8=PAGE FORWARD	F10=PAGE LEFT. F11=PAGE RIGHT.

Fig. 18. PRINT MENU WINDOW.

```

1  /******                                                    */ 00030011 *
1  /* COMPANY: CALSPAN/AEDC                                     */ 00050011 *
1  /* SOURCE CODE NAME: PRINT                                 */ 00070011 *
1  /* AUTHOR: ROBERT F. DAVIS                                 */ 00090011 *
1  /* PURPOSE:                                                */ 00110011 *
1  /*   EXAMPLE CODE FOR USE A SAS SOUTHEASTERN USER GROUP.   */ 00120015 **
1  /*   USED BY CALSPAN PLANNING AND WORK CONTROL MENU SYSTEM */ 00130015 **
1  /*   TO ASSIGN A REMOTE PRINTER, NUMBER OF COPIES, AND SEND OUTPUT */ 00140011 *
1  /*   TO THE ASSIGNED PRINTER.                             */ 00150011 *
1  /******                                                    */ 00190011 *
2  %GLOBAL GPRINT GNUMCOPY;                                   00200009 *
2  DATA PRINTER;                                           00210009 *
5  LENGTH PRINTER $6 NUMCOPY $2;                             00220009 *
1  PRINTER = UPCASE('X1'); /* INPUT PARAMETERS */           00230009 *
1  NUMCOPY = 'X2'; /* FROM PRINT MENUS */                   00240009 *
3  IF (PRINTER EQ 'X1') THEN PRINTER = 'RMT107'; /* DEFAULT PRINTER */ 00250009 *
4  IF (NUMCOPY EQ 'X2') THEN NUMCOPY = '1'; /* DEFAULT # OF COPIES */ 00260009 *
4  CALL SYMPUT('GPRINT', PRINTER); /* STORE OPTIONS */     00270009 *
4  CALL SYMPUT('GNUMCOPY', NUMCOPY); /* IN GLOBAL COMMON */ 00280009 *
4  DM OUT 'AUTOPOP OFF'; /* IN GLOBAL COMMON */             00300009 *
2  RUN;                                                      00320009 *
1  FILENAME REPORT DEST=&GPRINT COPIES=&GNUMCOPY SYSOUT=A;  00330009 *
1  DM OUT 'PRTFILE REPORT'; PRINT REPLACE; FREE; CLEAR;    00340009 *
35 /* WRITE DISPLAY MANAGER OUTPUT */                       00350009 *
38 /* WINDOW TO PRINTER */                                  00360009 *
38 /* RELEASE PRINT BUFFER */                                00370009 *
38 /* AND CLEAR OUTPUT WINDOW. */                            00380009 *
1  FILENAME REPORT CLEAR; /* DEASSIGN PRINT FILE */         00390009 *
1  RUN;                                                       00400009 *
1  %INCLUDE LIB(SEUGWIN); /* RETURN TO CALLING MENU */      00410009 *
1  RUN;                                                       00420009 *
1  RUN;                                                       00430009 *

```

Fig. 19. CONCLUDED.