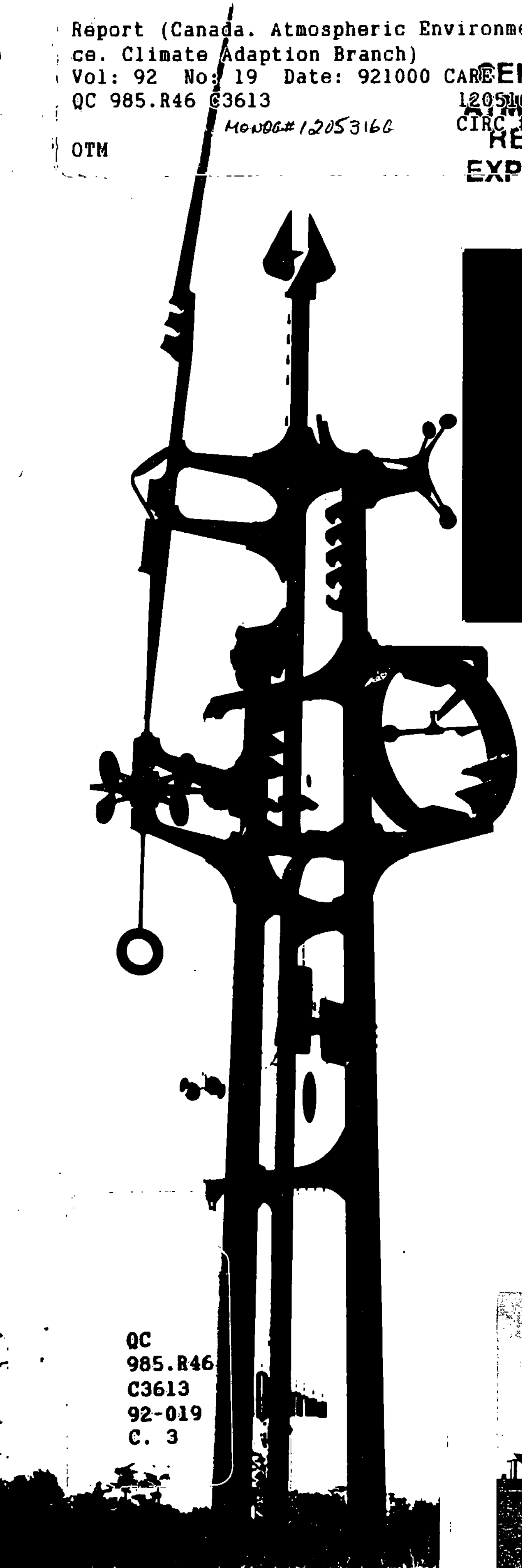
Environment Canada Imaging Cover Page

Report N.: 

SKP Box Number: 672572464

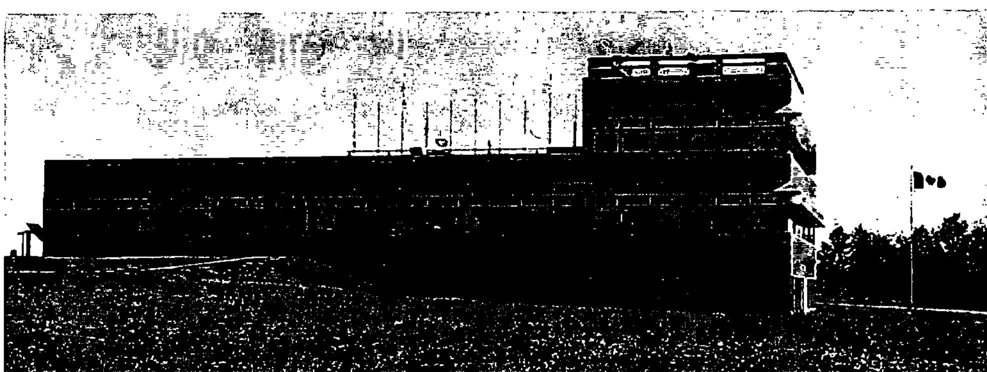
CENTRE FOR ATMOSPHERIC RESEARCH EXPERIMENTS

CENTRE DE RECHERCHE ATMOSPHÉRIQUE



Environment Canada / Environnement Canada

Atmospheric Environment Service / Service de l'environnement atmosphérique

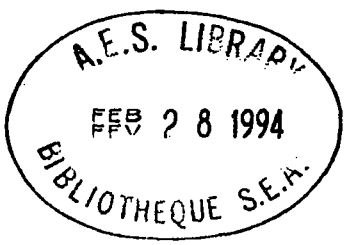


ECSDISP

ELECTRONIC CLIMATE SYSTEM DISPLAY

USER MANUAL

K. SINGH
CANADIAN CLIMATE CENTRE
ENVIRONMENT CANADA



REPORT NO.
CCAD-92-019(CARE)

OCTOBER 1992

ECSDISP

ELECTRONIC CLIMATE SYSTEM DISPLAY

USER MANUAL

K. SINGH
Bioclimate Adaptation Division
Adaptation Branch
Canadian Climate Centre
Atmospheric Environment Service

For additional copies contact:

Mr. D.C. MacIver
Bioclimate Adaptation Division
Canadian Climate Centre
4905 Dufferin Street
Donsview, Ontario, Canada
M3H 5T4

Phone: (416) 739-4391
Fax: (416) 739-4297

Mr. D.W. McNichol
Bioclimate Adaptation Division
Canadian Climate Centre
CARE RR#1
Egbert, Ontario, Canada
L0L 1N0

Phone: (705) 458-3321
Fax: (705) 458-3301

ACKNOWLEDGEMENT

The author wishes to express his gratitude towards the facilities that exist at the Centre for Atmospheric Experiments (CARE) to make the development of this software package possible. Special thanks to D.C. MacIver, D.W. McNichol, D. Phillips and T. Smith for their suggestions and guidance towards the Electronic Climate System Display software development.

1. INTRODUCTION

The Bioclimate Division (CCAD/B) has recently developed an integrated Electronic Climate System (ECS). The ECS a sophisticated system for monitoring, displaying and analysing hourly and daily climatological parameters. This manual deals with one portion of this complex but user friendly system, the extracting and analyzing of the archived data. This part of the ECS is almost entirely dependent on the set-up used for the ECS as a whole, therefore, make sure that all set-up files are in proper standing (see "THE ELECTRONIC CLIMATE SYSTEM - SET-UP PROCEDURES"). If the set-up files are properly defined, data extraction and analysis becomes a very simple, menu-driven routine.

2. The DATA PROCESSING AND GRAPHING main menu

To invoke the "DATA PROCESSING AND GRAPHING" main menu, select "INVOKE GRAPHING FACILITIES" from the "ARCHIVE" main menu. Once invoked, the main menu will display the following options:

1) FOREST NURSERY DEGREE DAYS
2) CALCULATE HEATING/COOLING DAYS
3) CALCULATE DEGREE DAYS
4) 24 HOUR QUALITY CONTROL GRAPHS
5) EXECUTE PC WINDROSE
6) WINDCHILL & HUMIDEX CONVERSIONS
7) SELECT DATA & INVOKE GRAPHING/STATISTICAL PKG
8) SOIL TEMPERATURE PROFILES
9) VERTICAL PROFILES
10) EDIT PROGRAM SET-UP FILES
11) RETURN TO ARCHIVE MAIN MENU
12) QUIT TO DOS

To execute any one of these options, simply highlight the required option by using the arrow keys and press enter, or press the key that precedes each option.

2.1 FOREST NURSERY DEGREE DAYS

This option will calculate total growing degree accumulations for a 0. 5, and 10 degree Celcius base using a predetermined start-up condition. The program will prompt the user for the input database file, output text file (do not enter extension - default is .txt), the desired station (again, selection is made by high-lighting the desired station and pressing enter), the start time and end time of analysis and the form of the time field (for example, the user might want only the Julian Day and not the year).

One should note that there are some fixed criteria that the models use. First, the minimum start-up date is May 11 (leap years are taken into account by the program), so to minimize processing time one should select May 10 for the start date. Similarly, the end-condition is October 15, therefore it is recommended that October 16 be the last date of the year. The program has default settings of the first day of the year for start date, and the current date for the end. One other built-in function, is that start-up procedure, which does not occur until five consecutive days have elapsed with a daily mean temperature of five degrees Celsius or greater. Also, the program keeps track of the last date executed. This feature lets the user "update" the values at a later date. By simply entering the start date as the end-date of the last execution, the program will continue processing from the previous executed date (note that the date must exist or else nursing units will re-zero).

The output file contains the following four fields:

1) TIME

2) STATION - Station ID

3) DATA_0 - The accumulated nursery units with a zero degree celsius base.

4) DATA_5 - As DATA_0 using five degree celsius base.

5) DATA_10 - As DATA_0 using 10 degree celsius base.

2.2 CALCULATE HEATING/COOLING DAYS

This option prompts the user with the same set of menus as "FOREST NURSERY DEGREE DAYS", however, the nature of calculations are different. This program will calculate the daily mean temperature for each day. For any day that is above 18 degrees Celsius, the mean temperature will be added to the cumulative cooling degree day total. Similarly, any mean temperature below 18 degrees Celcius will be added to the cumulative heating degree day totals. As in the preceeding selection, one can update the values from the last execution date. The output file contains the following fields:

1) TIME

2) STATION

3) HEATING - the accumulated heating degree days.

4) COOLING - the accumulated cooling degree days.

2.3 CALCULATE CORN HEAT UNITS (CHU)

This option calculates the corn heat units based on the equation: corn heat=(1.8*A+(3.33-.084*B)*B)/2 where A = (minimum daily temperature - 4.4) and B = (maximum day temp +10.0). The CHU value has the same start-up and ending conditions as "FOREST NURSERY DEGREE DAYS". The output file contains the following fields:

1) TIME

2) STATION

3) DAYMAX - maximum climate day temperature

4) DAYMIN - minimum climate day temperature

5) CORN - the corn heat calculated for the given day

6) TCORN - the total corn heat units accumulated to date

2.4 CALCULATE DEGREE DAYS

This option is the same as "FOREST NURSERY DEGREE DAYS" except that there is no start-up or end conditions.

2.5 24 HOUR QUALITY CONTROL GRAPH

A graph for the desired date (entered in a pop-up screen) is produced for each station that the ARCHIVE contains. For each station, curves for temperature, battery voltage, relative humidity, global solar radiation and wind direction as well as bar graphs for precipitation and wind speed, are produced. The user, through "EDIT PROGRAM SET-UP FILES" (see below), can select which observations to plot for each sensor type (ie., one can plot maximum, minimum and average temperatures all on the same graph). The program will graph one station at a time and will proceed to the next station after twenty seconds until all stations have been displayed. See Appendix B for graph requirements and interpretation.

2.6 EXECUTE PC WINDROSE

When selected, this option will prompt the user for the source and destination files (ie., the same as the "FOREST NURSERY DEGREE DAYS"), as well as the start and end dates for analysis. The PC WINDROSE SET-UP file stores all the criteria for the desired selection of data. To choose these parameters please refer to "PC WINDROSE" (Report No. CCAD-92-012) by J. Chin. The parameters may then be entered using the "EDIT PROGRAM SET-UP FILES".

Once the source and destination files along with the time is entered, the data will automatically be extracted and sent to the destination file, in the correct format for PC WINDROSE. When data extraction is completed, PC WINDROSE is executed.

2.7 WINDCHILL & HUMIDEX CONVERSIONS

Not completed as of documenting (SEPT 3 92)

2.8 SELECT DATA & INVOKE GRAPHING/STATISTICAL PKG

This is essentially the same data extraction procedure as in the "ARCHIVE" main menu (see "THE ELECTRONIC CLIMATE SYSTEM - AES_SYS) but with one added feature. After the desired data have been extracted, the program will first prompt the user to enter a date type (ie., YYYYDDDHH, YYYYDDD, etc). Once entered, the program will modify the TIME field from the default "YYYYDDDHH" to the selected one. After completing this task the user will be prompted again for the desired fields and logical relations. For example, if one wishes the output to contain the time, height, and temperature, simply replace the "F" with a "T" in those fields (first column). This will produce an output file with the time, height code, and the data value. However, if one desired the actual height (ie 1.5 m) instead of the height code (ie height code of "7") then enter a "T" in place of the "F" in the second column in the same row as height. The output file will be in text format with extension ".TXT"

2.9 SOIL TEMPERATURE PROFILES

This option will produce a text file with four columns as listed below:

1) RECORD NUMBER

2) TIME

3) DEPTH (OR HEIGHT)

4) TEMPERATURE (OR DATA VALUE)

These parameters can be imported directly into various contouring packages.

In order to create a more uniform grid, this program interpolates linearly between two sensor probes, creating three additional points between the two data points. By doing this, there are more points in the y-direction thus making a better database for the contouring software. Without this interpolation, it was noted that the contouring packages put too much emphasis on the x-direction, resulting in distorted mapping.

2.10 VERTICAL PROFILES

The VERTICAL PROFILES option produces temperature profiles (other parameter such as relative humidity and wind speed could also be profiled) using all the available height levels for the selected station. Through the set-up file (see EDIT PROGRAM FILES below), one can select the desired hours for which the profiles are to be plotted to a maximum of 12 profiles. For each profile, a file with the extension Vxx.TXT will be produced. If, for example, one selects the output file (on the output file pop-up) as "DAY1234" and selected hours 2, 5 and 10 to be profiled, the output files "DAY12V2.TXT", "DAY12V5.TXT" and "DAY12V10.TXT" will be created. Note that the output file name is truncated to five characters.

Once the files have been produced, a display of the profiles is created. If for any reason, one wishes to use another graphing packages, the files can be exported as ASCII files into the desired graphing package. (For more information see Appendix C)

2.11 EDIT PROGRAM SET-UP FILES

All of the previous option use "set-up" files in order to minimize user input when selecting out data or producing graphs such as the "24 HOUR QUALITY CONTROL GRAPHS". By selecting the "EDIT PROGRAM SET-UP FILES" option the user will be able to fine tune the system to his/her requirements. The user will first highlight the desired program to edit from the following:

1) 24 HOUR GRAPHS

2) WINDROSE SET-UP

3) FORMS SET-UP

4) SOIL TEMP PROFILE SET-UP

5) VERTICAL PROFILE SET-UP

Once a slection has been made, the editor will display one record of the chosen file at a time. To change a field use the arrow keys to position the cursor and press the F10 key. This keystroke will either produce a popup or highlight the current field (referred to as the edit mode). If a pop-up is displayed on the screen, simply highlight the desired field and press <enter>. If there is no pop-up, the field requires an input to which there is no code relation (ie when entering the name of a file). In this case, type in the desired value of cahracter string and press enter. Once the selection has been made (through pop-up or highlight) the user exited from edit mode. It is not possible to change a field without key-stroking the F10 key. If the Esc key is pressed while in edit mode the changes for the current field are not saved.

To continue to the next record, hit the pg-dn key. Like-wise, the pg-up key brings the user to the previous record. If the user is not in edit mode, the Esc will exit to the GRAPHICS main menu. All changes up to that point will be saved. For more details on individual set-up files.

see Appendix A.

APPENDIX A

A.1. 24 HOUR GRAPHS SET-UP

OUTFILE	SENSOR	MAKE	HEIGHT	OBSER1	OBSER2	OBSER3	OBSER4	ARRAY	DERIVED	DESCRIPT
ARCIVE										INPUT DATABASE NAME ONLY
STAFILE.TXT										STATION NAME FILE NAME ONLY
JDATE.TXT										JULIAN DATE FILE NAME ONLY
TEMP24.TXT	2	0	7	5	9	A	0	3	0	TEMPERATURE CRITERIA
BATT24.TXT	z	0	0	0	0	0	0	3	0	BATTERY
HUMID24.TXT	3	0	7	5	9	A	0	3	0	HUMIDITY CRITERIA
PRECIP24.TXT	4	0	7	8	0	0	0	4	0	PRECIPITATION (1 OBSER)
RAD24.TXT	5	0	7	8	0	0	0	3	0	RADIATION (MAX OF 2 OBSER)
WINDS24.TXT	7	0	9	5	9	0	0	3	0	WIND SPEED CRITERIA (MAX OF 2 OBSER)
WINDD24.TXT	6	0	9	5	0	0	0	3	0	WIND DIRECTION CRITERIA (MAX OF 2 OBSER)

Under OUTFILE these are the file used:

1) ARCHIVE file name.
2) Station file name used by GRAPH4.
3) Julian date file name used by GRAPH4.
4) Temperatures data file name used by GRAPH4.
5) Battery data file name.
6) Humidity data file name.
7) Precipitation data file name.
8) Radiation data file name.
9) Wind speed data file name.
10) Wind direction data file name.

The output data that is produced by selecting "24 HOUR QUALITY CONTROL GRAPHS" will be assigned filenames that have entered in the above selections #2-#10.

For files #4-#10 the user must input the SENSOR, MAKE, HEIGHT, OBSER1, OBSER2, OBSER3, OBSER4, ARRAY and DERIVED paramters. The SENSOR and MAKE fields correspond to the sensor type and make respectively. The HEIGHT field corresponds to the instrument height (or depth). The ARRAY field specifies which array in the ARHIVE to use. OBSER1 to OBSER4 are the various obervations that are to plotted. Finally, the DERIVED field indicates in this is a calculated field.

A.2. WINDROSE SET-UP

ARRAY	SENSOR	MAKE	HEIGHT	OBSER	DERIVED	DESCRIP
8	2	0	7	N	0	MAXIMUM TEMPERATURE
8	2	0	7	O	0	MINIMUM TEMPERATURE
8	3	0	7	N	0	MAX. RELATIVE HUMIDITY
8	3	0	7	O	0	MIN. RELATIVE HUMIDITY
8	7	0	9	N	0	MAX. 10m WIND SPEED
8	7	0	9	J	0	MEAN 10m WIND SPEED
8	7	0	8	J	0	MAX. 3m WIND SPEED
8	7	0	8	J	0	MEAN 3m WIND SPEED
8	6	0	9	J	0	MAX 10m WIND SPEED
8	6	0	9	J	0	WIND DIR @ TIME OF MAX. 10m WS

These paramters are similar to those of A.1. The descriptions on the right side indicate what is required.

A.3. FORMS SET-UP

FIELD_NAME	ARRAY	SENSOR	MAKE	HEIGHT	OBS_CODE	DERIVED	HOURS
MAX08	5	2	0	7	G	0	8
MIN08	5	2	0	7	H	0	8
RSET08	5	2	0	7	D	0	8
RAIN08	5	4	0	7	F	0	8
MAX16	5	2	0	7	G	0	16
MIN16	5	2	0	7	H	0	16
RSET16	5	2	0	7	D	0	16
RAIN16	5	4	0	7	F	0	16

Forms set-up is used by all the degree accumulation models (first 4 selections on the GRAPHICS main menu) and CLIMATE FORMS option on the ARCHIVE main menu. The MAX08, MIN08, RSET08, RAIN08 rows are used for the first observation on a given climate day while the rest are used for the second set of observations. The HOURS field indicates the time of observation.

SENSOR	MAKE	ARRAY	DERIVED	DESCRIPT	HOUR1	HOUR2	HOUR3	HOUR4	HOUR5	HOUR6	HOUR7	HOUR8	HOUR9	HOUR10	HOUR11	HOUR12
2	1	3	0	ENTER DESIRED HOURS IN PROCEEDING FIELDS	4	8	12	16	20	24	0	0	0	0	0	0

APPENDIX B

B.1. INTRODUCTION

GRAPH4 is a FORTRAN code that generates graphs for temperatures, relative humidity, wind direction and speed, precipitation, solar radiation and battery condition for a twenty-four hour scientific day. There are options to plot up to four different observations for each of the graphs (for example, one can plot maximum, minimum and instantaneous temperature).

GRAPH4 can be used on any IBM compatible personal computer with a minimum EGA graphics card. This program code was written and compiled using Microsoft FORTRAN V5.0. The data required for GRAPH4 can be created using a text editor, but for maximum efficiency, it is suggested that the DBASE code "EXTRACT" or "EXTRACT1" be used to create the DATA files.

B.2. DATA REQUIREMENTS AND PREPARATION

To execute GRAPH4, nine data files and one set-up file (all in text format) must exist as listed:

1) STAFILE.TXT ** file containing station names

2) JDATE.TXT ** file containing Julian date

3) TEMP24.TXT ** file containing temperatures

4) BATT24.TXT ** file containing battery voltages

5) HUMID24.TXT ** file containing relative humidity values

6) PRECIP24.TXT ** file containing precipitation values

7) RAD24.TXT ** file containing radiation values

8) WINDS24.TXT ** file containing wind speeds

9) WINDD24.TXT ** file containing wind directions

10) GRAFSU.TXT ** set-up file containing the names of data files

NOTE: The filenames of #1-#9 can be changed to any desired one (see Appendix A). The DBASE programs EXTRACT and EXTRACT1 will produce all of the above files, requiring no user intervention except possibly to enter the date of the required data (this of course is assuming that a database exists).

B.2.1 JDATE.TXT

This file simply contains the Julian date, for example 1992235 signifies year 1992, day 235.

B.2.2 STAFILE

This file contains the names of the available climate stations (one name per line).

B.2.3 TEMP24,BATT24,...WINDD24

These files contain data as described above. However, there is more than one field in these files. The first field contains the hour of observation (1-24). The second field contains the value associated with the graph type (ie., for temperature there would be a numeric value signifying the temperature at the given hour). The third field contains the station name. Last is the field containing the observation description (ie., maximum, average, etc.). Each file must be sorted by station, then time.

B.2.4 GRAFSU

This file contains the names of the data files #3-#9 above, in the same order.

B.3. EXECUTING GRAPH4

To execute GRAPH4 simply type "GRAPH4" and press enter. The program will read the data and display all of the various graphs, one station at a time, every twenty seconds, until finished.

B.4. INTERPRETING THE GRAPHS

When GRAPH4 is executed, there will be four sections on the screen, as listed:

1) TEMPERATURES & BATTERY COND.

2) RELATIVE HUMIDITY

3) PRECIPITATION & SOLAR RADIATION

4) WIND SPEED & DIRECTION

Each section will have an x-axis displaying hours (1-24) and a y-axis displaying either temperature (degrees), relative humidity (%), for precipitation (mm) or wind speed (km/h). However sections 1), 3) and 4) will have a secondary y-axis corresponding to battery voltages (V), solar radiation (kJ/m^2) and wind direction (degrees) respectively.

In the lower corners of each graph there will be a colour coded observation description. For example, in the TEMPERATURE & BATTERY COND. graph there might be a red curve and a white curve with a red coloured observation "average" displayed in the lower left corner and a white coloured observation "instantaneous" in the lower right corner. This would indicate that the red curve corresponds to the left y-axis and therefore an "average" temperature curve. Similarly, the white curve would indicate an "instantaneous" battery voltage.

NOTE: The "TEMPERATURE AND BATTERY COND." graph has a blue line at 0 degrees Celsius, indicating the freezing point of water. In the "WIND SPEED & DIRECTION" graph, there is a limit of two observations (this is because the graph is a bar type). Similarly for the "PRECIPITATION & SOLAR RADIATION", there can be only one type of observation for precipitation.

APPENDIX C

VERTGRAF.EXE is a FORTRAN source code that has the same system requirements as GRAPH4.EXE. VERTGRAF requires one set-up file with a name VFILES.TXT to execute properly (this file is automatically produced when selecting "VERTICAL PROFILES"). Although this file is automatically produced through the ECS, one may wish to execute this program outside the ECS. If this is the case the follwing rules must be followed:

1) In the file VFILES.TXT one must specify the data files being used (one datafile per line starting from row 2, column 2).

2) Each datafile must contain 3 fields starting from row 2, column 2. The first field is the time field (ie 199213502 or simply 02) indicating at least the hour. The second field is the height field, indicating the height of the instrument (ie 1.5). The third field contains the assocaited data value (ie 27.6).

The graph is simple to interpret. Each profile is colour coded, with the label being dislayed on the left side of the axis.

```
C     ************************************************************
C     *  PROGRAM GRAPH4                                          *
C     *  SEPARATES SCREEN INTO FOUR QUADRANTS AND GRAPHS THE     *
C     *  FOLLOWING:                                              *
C     *     QUADRANT1- GRAPH OF TEMPERATURE AND BATT VOLTS VS TIME *
C     *     QUADRANT2- GRAPH OF R.H. VS TIME                     *
C     *     QUADRANT3- GRAPH OF PRECIP & SOLRA RADIATION VS TIME *
C     *     QUADRANT4- GRAPH OF WIND SPEED AND DIRECTION VS TIME *
C     *                                                          *
C     *     FILES NEEDED:   GRAFSU.TXT  (SET-UP TEXT FILE)       *
C     *                                                          *
C     *  WRITTEN JULY 1992    K. SINGH                           *
C     ************************************************************

      INCLUDE 'c:\fortran\include\FGRAPH.FI'
      INCLUDE 'c:\fortran\INCLUDE\FGRAPH.FD'
      PARAMETER (NSTAT=10)
      PARAMETER (NFONTS=1)
      LOGICAL*1 UPTIM
      INTEGER*2 DLINE
      INTEGER*4 DLINE4
      CHARACTER*30 NEWSTATION(NSTAT),OLDSTATION
      CHARACTER*10 OPTIONS(NFONTS),JDATE
      CHARACTER*20 LIST
      CHARACTER*30 GTITLE
      CHARACTER*64 FONTPATH,JULIAN,TEMPER,BATTER,HUM,
     1             PRECIPI,RADIAT,WINDSP,WINDDIR,STATS
      RECORD /XYCOORD/ XY
      DATA OPTIONS    /'t''roman'*/

C     ****SET VIDEO MODE TO EGA COLOR (640 X 340)
      DLINE=SETVIDEOMODE($ERESCOLOR)
C     ****SET FONT LIBRARY FOR USE
      FONTPATH='ROMAN.FON'
      DLINE=REGISTERFONTS(FONTPATH)
      DLINE4=SETBKCOLOR($BLUE)
      CALL CLEARSCREEN($GCLEARSCREEN)
      CALL MOVETO(320,0,XY)
      DLINE=LINETO(320,349)
      CALL MOVETO(0,174,XY)
      DLINE=LINETO(639,174)
      ILEFT=0
      I=0

C  ************ READ GRAPHING SET-UP PARAMETERS ***************
      OPEN (5,FILE='GRAFSU.TXT',ACCESS='SEQUENTIAL',
     1           STATUS='OLD')
      READ (5,'(A)',END=10) FILLER
      READ (5,'(A)',END=10) STATS
      READ (5,'(A)',END=10) JULIAN
      READ (5,'(A)',END=10) TEMPER
      READ (5,'(A)',END=10) BATTER
      READ (5,'(A)',END=10) HUM
      READ (5,'(A)',END=10) PRECIPI
      READ (5,'(A)',END=10) RADIAT
      READ (5,'(A)',END=10) WINDSP
      READ (5,'(A)',END=10) WINDDIR
C     READ (5,*,END=10) IHR,IMIN
C   ********************************************************
10    CLOSE (5)
C      DLINE=REGISTERFONTS(FONTPATH)
      OPEN (5,FILE=STATS,ACCESS='SEQUENTIAL',
     1           STATUS='OLD')
      DO WHILE (.NOT. EOF(5))
        READ (5,121,END=30) NEWSTATION(I)
        I=I+1
      END DO
30    CLOSE (5)
121   FORMAT(1X,A30)
      OPEN(6,FILE=JULIAN,ACCESS='SEQUENTIAL',
     1           STATUS='OLD')
      READ(6,113,END=20) JDATE
      CLOSE (6)
20    DO WHILE (ILEFT .LT. (I-1))
         ILEFT=ILEFT+1
         OLDSTATION=NEWSTATION(ILEFT)
         CALL TEMPERATURE(OLDSTATION,TEMPER,BATTER)
         CALL HUMIDITY(OLDSTATION,HUM)
         CALL PRECIP(OLDSTATION,PRECIPI,RADIAT)
         CALL WIND(OLDSTATION,WINDSP,WINDDIR)
         LIST=OPTIONS(1)//'h12w8'
         CALL SETVIEWPORT(120,337,520,349)
         CALL CLEARSCREEN($GVIEWPORT)
         DLINE=SETFONT(LIST)
         GTITLE='DATE: '
         CALL MOVETO(0,0,XY)
         CALL OUTGTEXT(GTITLE)
         GTITLE=JDATE
         CALL MOVETO(48,0,XY)
         CALL OUTGTEXT(GTITLE)
         GTITLE='  STATION: '
         CALL MOVETO(104,0,XY)
         CALL OUTGTEXT(GTITLE)
         GTITLE=OLDSTATION
         CALL MOVETO(192,0,XY)
         CALL OUTGTEXT(GTITLE)
         CALL GETTIM (IHR,IMIN,ISEC,I100TH)
         ISEC1=MOD(ISEC+20,60)
         IF (ISEC1 .LT. ISEC) THEN
           IMIN1=MOD(IMIN+1,60)
           IF (IMIN1 .LT. IMIN) THEN
              IHR1=MOD(IHR+1,24)
           ENDIF
         ELSE
           IHR1=IHR
           IMIN1=IMIN
         ENDIF
```

```
          UPTIM = .FALSE.
          DO WHILE (.NOT. UPTIM)
            CALL GETTIM (IHR,IMIN,ISEC,I100TH)
            IF (IHR .GE. IHR1) THEN
               IF (IMIN .GE. IMIN1) THEN
                 IF (ISEC .GE. ISEC1) THEN
                    UPTIM = .TRUE.
                 ENDIF
               ENDIF
            ENDIF
          END DO
       END DO
       DLINE=SETVIDEOMODE($DEFAULTMODE)
110    FORMAT(1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2)
111    FORMAT(1X,A4,1X,A4,1X,A4,1X,A4)
112    FORMAT(1X,A1,1X,A1,1X,A1,1X,A1)
113    FORMAT(/1X,A7)
115    FORMAT(1X,A$)
       END


       SUBROUTINE TEMPERATURE (OLDSTATION,TEMPER,BATTER)

C  ***************************************************
C  ***  SUBROUTINE TO PLOT TEMPERATURES & BATTERY CONDITION ***
C  ***************************************************

       INCLUDE 'c:\fortran\include\FGRAPH.FD'
       PARAMETER (NFONTS = 1)
       INTEGER*2 DLINE,ICOLOR
       INTEGER*4 SCOLOR,IFOS
       CHARACTER*64 GTITLE(NFONTS),TEMPER,BATTER
       CHARACTER*5 GNUM
       CHARACTER*20 LIST
       CHARACTER*10 OPTIONS(NFONTS)
       CHARACTER*30 STATION,OBSCODE,OLDCODE,OLDSTATION
       DOUBLE PRECISION TEMP(25),TIME1(25),VOLT(25)
       RECORD /WXYCOORD/ WXY

       DATA OPTIONS /'t'roman'"/

C   ************** SET UP TEMPERATURE AXIS & PLOT POINTS ******

       IFONT=1
       LIST=OPTIONS(IFONT)//'h12w8'
       DLINE=SETFONT(LIST)
       OPEN (1, FILE=TEMPER,ACCESS = 'SEQUENTIAL',
      1          STATUS='OLD')
       CALL SETVIEWPORT(0,0,319,173)
       SCOLOR=SETCOLOR(14)
       CALL CLEARSCREEN($GVIEWPORT)
       DLINE=SETWINDOW (.TRUE.,-5.,-40.,28.,50.)
       CALL MOVETO_W(1.,48.,WXY)
       GTITLE= "TEMPERATURES & BATTERY COND."
       CALL OUTGTEXT(GTITLE(IFONT))
       SCOLOR=SETCOLOR(9)
       CALL MOVETO_W(0.,0.,WXY)
       DLINE=LINETO_W(24.,0.)
       SCOLOR=SETCOLOR(14)
       CALL MOVETO_W(0.,-25.,WXY)
       DLINE=LINETO_W(0.,40.)
       CALL MOVETO_W(0.,-25.,WXY)
       DLINE=LINETO_W(24.,-25.)

       LIST=OPTIONS(IFONT)//'h8w6'
       DLINE=SETFONT(LIST)
       CALL MOVETO_W(-4.8,5.,WXY)
       GTITLE="deg C"
       CALL OUTGTEXT(GTITLE(IFONT))
       CALL MOVETO_W(10.,-32.,WXY)
       GTITLE="Hour"
       CALL OUTGTEXT(GTITLE(IFONT))
       TICX=2.
       TICY=-25.
       DO WHILE (TICX .LT. 25.)
          CALL MOVETO_W(TICX,-26.,WXY)
          DLINE=LINETO_W(TICX,-24.)
          ITICX=INT(TICX)
          write(GTITLE,50) ITICX
          CALL MOVETO_W(TICX-.5,-27.,WXY)
          CALL OUTGTEXT(GTITLE(IFONT))
          TICX=TICX + 2.
       END DO
       DO WHILE (TICY .LT. 40.)
          CALL MOVETO_W(-.2,TICY,WXY)
          DLINE=LINETO_W(.2,TICY)
          ITICY=INT(TICY)
          WRITE(GNUM,60) ITICY
          CALL MOVETO_W(-2.5,(TICY+3.),WXY)
          CALL OUTGTEXT(GNUM)
          TICY=TICY+5.
       END DO

       ICOUNT = 1
       IFOS=-41
       IFORMAT=1
       OLDCODE ="0"
       ICOLOR=14

       DO WHILE (.NOT. EOF(1))
         READ (1,11),END=10) TIME1(ICOUNT),TEMP(ICOUNT),STATION,
      1       OBSCODE
         IF (STATION .EQ. OLDSTATION) THEN
           IF (OBSCODE .EQ. OLDCODE)  THEN
             CALL MOVETO_W (TIME1(ICOUNT-1),TEMP(ICOUNT-1),WXY)
             DLINE=LINETO_W (TIME1(ICOUNT),TEMP(ICOUNT))
           ELSE
```

```
            ICOLOR=ICOLOR-1
            SCOLOR=SETCOLOR(ICOLOR)
            TIME1(1)=TIME1(ICOUNT)
            TEMP(1)=TEMP(ICOUNT)
            OLDCODE = OBSCODE
            ICOUNT=1
            GTITLE=OLDCODE
            IFORMAT=IFORMAT+1
            IPOS=IPOS+5
            CALL MOVETO_W(-5,IPOS,WXY)
            CALL OUTGTEXT(GTITLE)
          ENDIF
        ELSE
          ICOUNT=1
        ENDIF
        ICOUNT = ICOUNT + 1
      END DO
10    CLOSE (1)

C     ********** SET UP BATTERY AXIS & PLOT POINTS *************

      DLINE = SETWINDOW(.TRUE.,-5.,0.,28.,20.)

      SCOLOR=SETCOLOR(15)
      CALL MOVETO_W(24.,9.,WXY)
      DLINE=LINETO_W(24.,16.)
      TICY=9
      CALL MOVETO_W(26.4,11.,WXY)
      GTITLE='V'
      CALL OUTGTEXT(GTITLE)
      DO WHILE (TICY .LT. 17.)
         CALL MOVETO_W(23.8,TICY,WXY)
         DLINE=LINETO_W(24.2,TICY)
         ITICY=INT(TICY)
         WRITE(GNUM,50) ITICY
         CALL MOVETO_W(24.5,(TICY+.5),WXY)
         CALL OUTGTEXT(GNUM)
         TICY=TICY+1.
      END DO

      OPEN (1,FILE=BATTER,ACCESS='SEQUENTIAL',
     1        STATUS='OLD')


      ICOUNT=1
      OLDCODE='0'
      DO WHILE (.NOT. EOF(1))
        READ (1,111,END=20) TIME1(ICOUNT),VOLT(ICOUNT),STATION,OBSCODE
C       write(6,*) count,time1(icount),temp(icount),station
        IF ((STATION .EQ. OLDSTATION) .AND. (OBSCODE .EQ. OLDCODE)) THEN
          CALL MOVETO_W (TIME1(ICOUNT-1),VOLT(ICOUNT-1)/2,WXY)
          DLINE=LINETO_W (TIME1(ICOUNT),VOLT(ICOUNT)/2)
        ELSE
          SCOLOR=SETCOLOR(15)
          OLDCODE=OBSCODE
          VOLT(1)=VOLT(ICOUNT)
          TIME1(1)=TIME1(ICOUNT)
          GTITLE=OBSCODE
          CALL MOVETO_W(23.,-32,WXY)
          CALL OUTGTEXT(GTITLE)
          ICOUNT=1
        ENDIF
      ICOUNT = ICOUNT + 1
      END DO
20    CLOSE (1)

50    FORMAT(i2.2)
60    FORMAT(I3)
111   FORMAT(1X,D9.0,1X,D9.2,1X,A30,1X,A8)
      RETURN
      END


      SUBROUTINE HUMIDITY(OLDSTATION,HUM)

C     ************************************************
C     ***       SUBROUTINE FOR PLOTING HUMIDITY      ***
C     ************************************************

      INCLUDE 'C:\FORTRAN\INCLUDE\FGRAPH.FD'
      PARAMETER (NFONTS=1)
      INTEGER*2 DLINE,ICOLOR
      INTEGER*4 SCOLOR,IPOS
      CHARACTER*64 GTITLE,LIST,HUM
      CHARACTER*5 GNUM
      CHARACTER*10 OPTIONS(NFONTS)
      CHARACTER*30 STATION,OBSCODE,OLDCODE,OLDSTATION
      DOUBLE PRECISION TIME(25),RH(25)
      RECORD /WXYCOORD/ WXY

      DATA OPTIONS /'t'roman'*/
      OPEN (2,FILE=HUM,ACCESS='SEQUENTIAL',
     1        STATUS='OLD')
      CALL SETVIEWPORT(321,0,639,173)
      SCOLOR=SETCOLOR(14)
      CALL CLEARSCREEN($GVIEWPORT)
      DLINE=SETWINDOW(.TRUE.,-5.,-30.,27.,130.)
      LIST=OPTIONS(1)//'h12w8'
      DLINE=SETFONT(LIST)

      GTITLE='RELATIVE HUMIDITY'
      CALL MOVETO_W(1.,125.,WXY)
      CALL OUTGTEXT(GTITLE)
      SCOLOR=SETCOLOR(14)
      CALL MOVETO_W(0.,0.,WXY)
      DLINE=LINETO_W(24.,0.)
      CALL MOVETO_W(0.,0.,WXY)
      DLINE=LINETO_W(0.,100.)
```

```
      LIST=OPTIONS(NFONTS)//'h8w6'
      DLINE=SETFONT(LIST)
      CALL MOVETO_W(-4.5,55.,WXY)
      GTITLE='%'
      CALL OUTGTEXT(GTITLE)
      CALL MOVETO_W(10.,-13.,WXY)
      GTITLE='Hour'
      CALL OUTGTEXT(GTITLE)
      TICX=2.
      TICY=0.
      DO WHILE (TICX .LT. 25.)
         CALL MOVETO_W(TICX,1.,WXY)
         DLINE=LINETO_W(TICX,-1.)
         ITICX=INT(TICX)
         write(GNUM,50) ITICX
         CALL MOVETO_W(TICX-.5,-2.,WXY)
         CALL OUTGTEXT(GNUM)
         TICX=TICX + 2.
      END DO
      DO WHILE (TICY .LT. 101.)
         CALL MOVETO_W(-.2,TICY,WXY)
         DLINE=LINETO_W(.2,TICY)
         ITICY=INT(TICY)
         WRITE(GNUM,60) ITICY
         CALL MOVETO_W(-2.5,(TICY+5.),WXY)
         CALL OUTGTEXT(GNUM)
         TICY=TICY+10.
      END DO

      ICOUNT=1
      ICOLOR=14
      IFORMAT=1
      IPOS=-30
      OLDCODE='0'
      DO WHILE (.NOT. EOF(2))
        READ (2,111,END=10) TIME(ICOUNT),RH(ICOUNT),STATION,OBSCODE
        IF (STATION .EQ. OLDSTATION) THEN
          IF (OLDCODE .EQ. OBSCODE) THEN
            CALL MOVETO_W (TIME(ICOUNT-1),RH(ICOUNT-1),WXY)
            DLINE=LINETO_W (TIME(ICOUNT),RH(ICOUNT))
          ELSE
            ICOLOR=ICOLOR-1
            SCOLOR=SETCOLOR(ICOLOR)
            RH(1)=RH(ICOUNT)
            TIME(1)=TIME(ICOUNT)
            OLDCODE=OBSCODE
            ICOUNT=1
            GTITLE=OBSCODE
            IFORMAT=IFORMAT+1
            IPOS=IPOS+8
            CALL MOVETO_W(-5,IPOS,WXY)
            CALL OUTGTEXT(GTITLE)
          ENDIF
        ELSE
          ICOUNT=1
        ENDIF
      ICOUNT=ICOUNT+1
      END DO
10    CLOSE(2)

50    FORMAT(i2.2)
60    FORMAT(I3)
111   FORMAT(1X,D9.0,1X,D9.2,1X,A10,1X,A8)

      RETURN
      END


      SUBROUTINE PRECIP(OLDSTATION,PRECIPI,RADIAT)

C     ***************************************************
C     ***  SUBROUTINE FOR PLOTTING PRECIPITATION & RADIATION  ***
C     ***************************************************

      INCLUDE 'C:\FORTRAN\INCLUDE\FGRAPH.FD'
      PARAMETER (NFONTS=1)
      INTEGER*2 DLINE
      INTEGER*4 SCOLOR
      CHARACTER*64 GTITLE,LIST,PRECIPI,RADIAT
      CHARACTER*5 GNUM
      CHARACTER*10 OPTIONS(NFONTS)
      CHARACTER*30  OBSCODE,OLDCODE,STATION,OLDSTATION
      DOUBLE PRECISION TIME(25),PREC(25),RAD(25)
      RECORD /WXYCOORD/ WXY

      DATA OPTIONS /'t'roman''/

C     ********* SET UP AXIS & PLOT PRECIPITATION VALUES  ********

      OPEN (3,FILE=PRECIPI,ACCESS='SEQUENTIAL',
     1        STATUS='OLD')
      CALL SETVIEWPORT(0,175,319,339)
      SCOLOR=SETCOLOR(14)
      CALL CLEARSCREEN($GVIEWPORT)
      DLINE=SETWINDOW(.TRUE.,-5.,-15.,28.,65.)
      LIST=OPTIONS(1)//'h12w8'
      DLINE=SETFONT(LIST)
      GTITLE='PRECIPITATION & SOLAR RAD.'
      CALL MOVETO_W(2.,62.,WXY)
      CALL OUTGTEXT(GTITLE)
      CALL MOVETO_W(0.,0.,WXY)
      DLINE=LINETO_W(24.,0.)
      CALL MOVETO_W(0.,0.,WXY)
      DLINE=LINETO_W(0.,50.)

      LIST=OPTIONS(1)//'h8w6'
      DLINE=SETFONT(LIST)
```

```
      CALL MOVETO_W(-4.8,27.,WXY)
      GTITLE="mm"
      CALL OUTGTEXT(GTITLE)
      CALL MOVETO_W(10,-8,WXY)
      GTITLE="Hour"
      CALL OUTGTEXT(GTITLE)
      TICX=2.
      TICY=0.
      DO WHILE (TICX .LT. 25.)
         CALL MOVETO_W(TICX,1.,WXY)
         DLINE=LINETO_W(TICX,-1.)
         ITICX=INT(TICX)
         write(GNUM,50) ITICX
         CALL MOVETO_W(TICX-.5,-2.,WXY)
         CALL OUTGTEXT(GNUM)
         TICX=TICX + 2.
      END DO
      DO WHILE (TICY .LT. 51.)
         CALL MOVETO_W(-.2,TICY,WXY)
         DLINE=LINETO_W(.2,TICY)
         ITICY=INT(TICY)
         WRITE(GNUM,60) ITICY
         CALL MOVETO_W(-2.5,(TICY+2.),WXY)
         CALL OUTGTEXT(GNUM)
         TICY=TICY+5.
      END DO

      ICOUNT=1
      OLDCODE="0"
      DO WHILE (.NOT. EOF(3))
        READ (3,111,END=10) TIME(ICOUNT),PREC(ICOUNT),STATION,OBSCODE
        IF (STATION .EQ. OLDSTATION) THEN
          IF (OLDCODE .EQ. OBSCODE) THEN
             DLINE=RECTANGLE_W($GFILLINTERIOR,TIME(ICOUNT-1),
     1       PREC(ICOUNT),TIME(ICOUNT),0.)
          ELSE
            SCOLOR=SETCOLOR(10)
            TIME(1)=TIME(ICOUNT)
            PREC(1)=PREC(ICOUNT)
            OLDCODE=OBSCODE
            ICOUNT=1
            GTITLE=OLDCODE
            CALL MOVETO_W(-5.,-10,WXY)
            CALL OUTGTEXT(GTITLE)
          ENDIF
        ELSE
          ICOUNT=1
        ENDIF
      ICOUNT=ICOUNT+1
      END DO
10    CLOSE(3)

C     ********** SET UP SOLAR RADIATION AXIS & PLOT VALUES ******

      OPEN (3,FILE=RADIAT,ACCESS='SEQUENTIAL',
     1        STATUS='OLD')
      SCOLOR=SETCOLOR(15)
      DLINE=SETWINDOW(.TRUE.,-5.,-1400.,28.,6000.)
      CALL MOVETO_W(24.,0.,WXY)
      DLINE=LINETO_W(24.,5000.)

      LIST=OPTIONS(1)//'h8w6'
      DLINE=SETFONT(LIST)
      CALL MOVETO_W(23.5,5600.,WXY)
      GTITLE="kJ/m**2"
      CALL OUTGTEXT(GTITLE)
      TICY=0.
      DO WHILE (TICY .LT. 5001.)
         CALL MOVETO_W(23.8,TICY,WXY)
         DLINE=LINETO_W(24.2,TICY)
         ITICY=INT(TICY)
         WRITE(GNUM,70) ITICY
         CALL MOVETO_W(24.4,(TICY+200.),WXY)
         CALL OUTGTEXT(GNUM)
         TICY=TICY+500.
      END DO

      ICOUNT=1
      OLDCODE="0"
      DO WHILE (.NOT. EOF(3))
        READ (3,111,END=20) TIME(ICOUNT),RAD(ICOUNT),STATION,OBSCODE
        IF (STATION .EQ. OLDSTATION) THEN
          IF (OLDCODE .EQ. OBSCODE) THEN
             CALL MOVETO_W(TIME(ICOUNT-1),RAD(ICOUNT-1),WXY)
             DLINE=LINETO_W(TIME(ICOUNT),
     1       RAD(ICOUNT))
          ELSE
            SCOLOR=SETCOLOR(15)
            TIME(1)=TIME(ICOUNT)
            PREC(1)=PREC(ICOUNT)
            OLDCODE=OBSCODE
            ICOUNT=1
            GTITLE=OLDCODE
            CALL MOVETO_W(23.,-800.,WXY)
            CALL OUTGTEXT(GTITLE)
          ENDIF
        ELSE
          ICOUNT=1
        ENDIF
      ICOUNT=ICOUNT+1
      END DO
20    CLOSE(3)


50    FORMAT(i2.2)
60    FORMAT(i3)
70    FORMAT(i4)
111   FORMAT(1X,D9.0,1X,D9.2,1X,A30,1X,A8)
```

```
      RETURN
      END


      SUBROUTINE WIND(OLDSTATION,WINDSP,WINDDIR)

C     ***************************************************
C     ***      SUBROUTINE TO PLOT WIND SPEED & DIRECTION      ***
C     ***************************************************

      INCLUDE 'C:\FORTRAN\INCLUDE\FGRAPH.FD'
      PARAMETER (NFONTS=1)
      INTEGER*2 DLINE
      INTEGER*4 SCOLOR
      CHARACTER*64 GTITLE,LIST,WINDDIR,WINDSP
      CHARACTER*5 GNUM
      CHARACTER*10 OPTIONS(NFONTS)
      CHARACTER*30 OBSCODE,OLDCODE,STATION,OLDSTATION
      DOUBLE PRECISION TIME(25),WD(25),WS(25)
      RECORD /WXYCOORD/ WXY

      DATA OPTIONS /'t'roman''/

C  ************** SET UP AXIS FOR WIND SPEED & PLOT VALUES ****

      OPEN (4,FILE=WINDSP,ACCESS='SEQUENTIAL',
     1       STATUS='OLD')
      CALL SETVIEWPORT(321,175,639,339)
      DLINE=SETCOLOR(5)
      CALL CLEARSCREEN($GVIEWPORT)
      DLINE=SETWINDOW(.TRUE.,-5.,-40.,28.,170.)
      LIST=OPTIONS(1)//'h12w8'
      DLINE=SETFONT(LIST)

      SCOLOR=SETCOLOR(14)
      GTITLE="WIND SPEED & DIRECTION"
      CALL MOVETO_W(1.,160.,WXY)
      CALL OUTGTEXT(GTITLE)
      CALL MOVETO_W(0.,0.,WXY)
      DLINE=LINETO_W(24.,0.)
      CALL MOVETO_W(0.,0.,WXY)
      DLINE=LINETO_W(0.,140.)
      LIST=OPTIONS(NFONTS)//'h8w6'
      DLINE=SETFONT(LIST)
      CALL MOVETO_W(-4.8,72.,WXY)
      GTITLE="km/h"
      CALL OUTGTEXT(GTITLE)
      CALL MOVETO_W(10.,-20.,WXY)
      GTITLE="Hour"
      CALL OUTGTEXT(GTITLE)
      TICX=2.
      TICY=0.
      DO WHILE (TICX .LT. 25.)
         CALL MOVETO_W(TICX,1.,WXY)
         DLINE=LINETO_W(TICX,-1.)
         ITICX=INT(TICX)
         write(GNUM,50) ITICX
         CALL MOVETO_W(TICX-.5,-2.,WXY)
         CALL OUTGTEXT(GNUM)
         TICX=TICX + 2.
      END DO
      DO WHILE (TICY .LT. 141.)
         CALL MOVETO_W(-.2,TICY,WXY)
         DLINE=LINETO_W(.2,TICY)
         ITICY=INT(TICY)
         WRITE(GNUM,60) ITICY
         CALL MOVETO_W(-2.5,(TICY+2.),WXY)
         CALL OUTGTEXT(GNUM)
         TICY=TICY+20.
      END DO

      ICOUNT=1
      IFORMAT=1
      IPOS=-38
      OLDCODE="0"
      IOBS=0
      DO WHILE (.NOT. EOF(4))
        READ (4,111,END=10) TIME(ICOUNT),WS(ICOUNT),STATION,OBSCODE
        IF (STATION .EQ. OLDSTATION) THEN
          IF (OLDCODE .EQ. OBSCODE) THEN
            DLINE=RECTANGLE_W($GFILLINTERIOR,TIME(ICOUNT-1)+RSET,
     1           WS(ICOUNT-1),TIME(ICOUNT-1)+RSET2,.5)
          ELSE
            WS(1)=WS(ICOUNT)
            TIME(1)=TIME(ICOUNT)
            IOBS=IOBS+1
            IF (IOBS .EQ. 1) THEN
               SCOLOR=SETCOLOR(12)
               RSET=.5
               RSET2=.9
            ENDIF
            IF (IOBS .EQ. 2) THEN
               SCOLOR=SETCOLOR(13)
               RSET=0
               RSET2=.4
            ENDIF
            OLDCODE=OBSCODE
            GTITLE=OLDCODE
            IFORMAT=IFORMAT+1
            IPOS=IPOS+11
            CALL MOVETO_W(-5.,IPOS,WXY)
            CALL OUTGTEXT(GTITLE)
            ICOUNT=1
          ENDIF
        ELSE
          ICOUNT=1
        ENDIF
```

```
      TICX=-30.
      TICY=0.
      DO WHILE (TICX .LT. 50.)
         CALL MOVETO_W(TICX,-.2,WXY)
         DLINE=LINETO_W(TICX,.2)
         ITICX=INT(TICX)
         write(GTITLE,60) ITICX
         CALL MOVETO_W(TICX-1.2,-.5,WXY)
         CALL OUTGTEXT(GTITLE(IFONT))
         TICX=TICX + 5.
      END DO
      DO WHILE (TICY .LT. 40.)
         CALL MOVETO_W(-30.3,TICY,WXY)
         DLINE=LINETO_W(-29.7,TICY)
         ITICY=INT(TICY)
         WRITE(GNUM,60) ITICY
         CALL MOVETO_W(-34.,(TICY+.5),WXY)
         CALL OUTGTEXT(GNUM)
         TICY=TICY+5.
      END DO

      IPOS=-4
      ICOLOR=14
      DO 222 IVHR=1,IR-1
      ICOUNT=1
      IF (FIL(IVHR) .EQ. ' ') THEN
        GOTO 20
      ENDIF
      OPEN (1,FILE=FIL(IVHR),ACCESS='SEQUENTIAL',STATUS='OLD')
C      READ (1,'(A)') SKIP
      READ (1,'(A)') SKIP
      DO WHILE (.NOT. EOF(1))
        READ (1,111,END=20) TIME1,TEMP(ICOUNT),HEIGHT(ICOUNT)
        IF (ICOUNT .GT. 1) THEN
           CALL MOVETO_W (TEMP(ICOUNT-1),HEIGHT(ICOUNT-1),WXY)
           DLINE=LINETO_W (TEMP(ICOUNT),HEIGHT(ICOUNT))
        ELSE
           ICOLOR=ICOLOR-1
           SCOLOR=SETCOLOR(ICOLOR)
           GTITLE=TIME1
           IPOS=IPOS+1.4
           CALL MOVETO_W(-44,IPOS,WXY)
           CALL OUTGTEXT(GTITLE)
        ENDIF
        ICOUNT = ICOUNT + 1
      END DO
20    CLOSE (1)
222   CONTINUE
50    FORMAT(i2.2)
60    FORMAT(I3)
111   FORMAT(1X,A9,1X,D9.2,1X,D9.2)
      RETURN
      END
```

```
C     ************************************************************
C     *  PROGRAM VERTGRAF                                        *
C     *       GRAPH OF VERTICAL PROFILE                          *
C     *                                                          *
C     *     FILES NEEDED:   GRAFVU.TXT  (SET-UP TEXT FILE)       *
C     *                                                          *
C     *  WRITTEN SEPT 1992    K. SINGH                           *
C     ************************************************************

      INCLUDE 'c:\fortran\include\FGRAPH.FI'
      INCLUDE 'c:\fortran\INCLUDE\FGRAPH.FD'
      PARAMETER (NSTAT=10)
      PARAMETER (NFONTS=1)
      INTEGER*1 DIAGMASK(8)
      INTEGER*2 DLINE
      INTEGER*4 DLINE4
      CHARACTER*10 OPTIONS(NFONTS)
      CHARACTER*20 LIST
      CHARACTER*30 GTITLE,FIL(12)
      CHARACTER*64 FONTPATH
C      COMMON FIL(12)
      RECORD /XYCOORD/ XY
      DATA OPTIONS     /'t'roman'*/
      DATA DIAGMASK / #93, #C9, #64, #B2, #59, #2C,
     1                #96, #4B /
C     ****SET VIDEO MODE TO EGA COLOR (640 X 340)
      DLINE=SETVIDEOMODE($ERESCOLOR)
C     ****SET FONT LIBRARY FOR USE
      FONTPATH='ROMAN.FON'
      DLINE=REGISTERFONTS(FONTPATH)

      DLINE4=SETBKCOLOR($BLUE)
      CALL CLEARSCREEN($GCLEARSCREEN)
      CALL SETFILLMASK(DIAGMASK)
      DLINE=SETCOLOR(3)
      DLINE=FLOODFILL(50,50,3)
C      CALL MOVETO(589,30,XY)
      DLINE=SETCOLOR(8)
      DLINE=RECTANGLE($GFILLINTERIOR,589,30,597,320)
      DLINE=RECTANGLE($GFILLINTERIOR,58,314,589,320)
      LIST=OPTIONS(1)//'h16w12'
      DLINE=SETFONT(LIST)
      CALL MOVETO(218,5,XY)
      DLINE=SETCOLOR(14)
      GTITLE="VERTICAL PROFILES"
      CALL OUTGTEXT(GTITLE)
      CALL TEMPERATURE
      READ(5,*) IWAIT
      DLINE=SETVIDEOMODE($DEFAULTMODE)
      END


      SUBROUTINE TEMPERATURE

C  ************************************************************
C  ***  SUBROUTINE TO PLOT HEIGHT VS TEMPERATURES           ***
C  ************************************************************

      INCLUDE 'c:\fortran\include\FGRAPH.FD'
      PARAMETER (NFONTS = 1)
      INTEGER*2 DLINE,ICOLOR
      INTEGER*4 SCOLOR,DLINE4
      CHARACTER*64 GTITLE(NFONTS)
      CHARACTER*5 GNUM,SKIP
      CHARACTER*20 LIST
      CHARACTER*10 OPTIONS(NFONTS)
      CHARACTER*30 TIME1,FIL(12)
      DOUBLE PRECISION HEIGHT(25),TEMP(25),IPOS
      RECORD /WXYCOORD/ WXY

      DATA OPTIONS /'t'roman'*/


C  ************ READ GRAPHING SET-UP PARAMETERS ***************
      OPEN (5,FILE='VFILES.TXT',FORM='FORMATTED',
     1          RECL=40,STATUS='OLD')
      READ (5,'(A)') SKIP
      DO 333 IR=1,13
        READ (5,'(A)',END=10) FIL(IR)
333   CONTINUE

C   ************************************************************
10    CLOSE (5)
C   ************** SET UP TEMPERATURE AXIS & PLOT POINTS ******

      IFONT=1

      CALL SETVIEWPORT(50,25,589,314)
      SCOLOR=SETCOLOR(14)
      CALL CLEARSCREEN($GVIEWPORT)
      DLINE=SETWINDOW (.TRUE.,-45.,-5.,50.,40.)
      CALL MOVETO_W(-30.,0.,WXY)
      DLINE=LINETO_W(45.,0.)
      CALL MOVETO_W(-30.,0.,WXY)
      DLINE=LINETO_W(-30.,35.)
      LIST=OPTIONS(IFONT)//'h12w8'
      DLINE=SETFONT(LIST)
      CALL MOVETO_W(-10,-2.,WXY)
      GTITLE="TEMPERATURE (deg C)"
      CALL OUTGTEXT(GTITLE(IFONT))
      CALL MOVETO_W(-44.,18.,WXY)
      GTITLE="HEIGHT (m)"
      CALL OUTGTEXT(GTITLE(IFONT))
      LIST=OPTIONS(IFONT)//'h10w6'
      DLINE=SETFONT(LIST)
```

```
      ICOUNT=ICOUNT+1
      END DO
10    CLOSE(4)

C     ******** SET UP AXIS FOR WIND DIRECTION & PLOT VALUES ****

      OPEN(4,FILE=WINDDIR,ACCESS='SEQUENTIAL',
     1         STATUS='OLD')
      SCOLOR=SETCOLOR(15)
      DLINE=SETWINDOW(.TRUE.,-5,-105.,28.,440.)
      CALL MOVETO_W(24.5,400.,WXY)
      GTITLE='deg'
      CALL OUTGTEXT(GTITLE)
      TICY=0
      CALL MOVETO_W(24.,0.,WXY)
      DLINE=LINETO_W(24.,360.)
      DO WHILE (TICY .LT. 370.)
         CALL MOVETO_W(23.8,TICY,WXY)
         DLINE=LINETO_W(24.2,TICY)
         ITICY=INT(TICY)
         WRITE(GNUM,60) ITICY
         CALL MOVETO_W(24.5,(TICY+10.),WXY)
         CALL OUTGTEXT(GNUM)
         TICY=TICY+30.
      END DO

      ICOUNT=1
      OLDCODE='0'
      SCOLOR=SETCOLOR(15)
      DO WHILE (.NOT. EOF(4))
        READ (4,111,END=20) TIME(ICOUNT),WD(ICOUNT),STATION,OBSCODE
        IF (STATION .EQ. OLDSTATION) THEN
          IF (OLDCODE .EQ. OBSCODE) THEN
            CALL MOVETO_W(TIME(ICOUNT-1),WD(ICOUNT-1),WXY)
            DLINE=LINETO_W(TIME(ICOUNT),WD(ICOUNT))
          ELSE
            TIME(1)=TIME(ICOUNT)
            WD(1)=WD(ICOUNT)
            ICOUNT=1
            OLDCODE=OBSCODE
            GTITLE=OLDCODE
            CALL MOVETO_W(23.,-80.,WXY)
            CALL OUTGTEXT(GTITLE)
          ENDIF
        ELSE
          ICOUNT=1
        ENDIF
        ICOUNT=ICOUNT+1
        END DO
20    CLOSE(4)

50    FORMAT(I2.2)
60    FORMAT(I3)
111   FORMAT(1X,D9.0,1X,D9.2,1X,A30,1X,A8)

      RETURN
      END
```

```
*************************************************************************************
* PROGRAM NAME:                 AES_OUT.PRG
*                               MAIN MENU
*                               DATA PROCESSING & GRAPHING FACILITY
* LAST CHANGED:                 SEPT 4, 92
* WRITTEN BY:                   DAVID PHILLIPS/K. SINGH
*************************************************************************************

PROCEDURE AES_OUT

   * Display menu and loop for choices
   DO PUBINIT
   mstrloop = .T.
   DO WHILE mstrloop
      DO inital && initalize color and file access parameters
      SET COLOR TO &c_normal.
      CLEAR
      DO title
      SET COLOR TO &c_pop.
      SET COLOR OF HIGHLIGHT TO &lt_blue.
      @ 8,12 TO 23,67 COLOR N/N
      ACTIVATE POPUP MAINOUT
   ENDDO
   CLOSE ALL
   CLEAR
RETURN
*========================== end of main procedure ==========================

PROCEDURE INITAL
   * Initailizing values
   DO PUBINIT  && ESTABLISH PROCECESS LOOP PARAMETERS
   DO PUBCOLOR && ESTABLISH COLOR PARAMETERS
   DO PUBFILE  && ESTABLISH FILE ACCESS PARAMETERS
   DO PUBPOPUP && ESTABLISH POPUP CONTROL PARAMETERS
   DO PUBCONT  && ESTABLISH SELECTION PARAMETERS
   mstrloop = .T.
   vdisk1 = GETENV("vdisk") && VIRTUAL DISK DRIVE
   IF ISALPHA(vdisk1)
      vdisk = LTRIM(RTRIM(vdisk1))
      IF vdisk = " "
           vdisk = "0"
      ENDIF
   ELSE
      vdisk = "0"
   ENDIF
   fileout = "           "
   filein = "ARCIVE.DBF"
   SET FULLPATH ON
   USE &filein
   pathin1  = LTRIM(RTRIM(DBF()))
   len1 = LEN(pathin1)
   pathin = LEFT(pathin1,len1-10)
   pathout = pathin
   SET FULLPATH OFF
   CLOSE DATABASES
   sta1 = ""
   arr1 = ""
   sen1 = ""
   mak1 = ""
   ht1  = ""
   obs1 = ""
   der1 = ""
   num1 = 0
   hr1  = 1
   hr2  = 24
   num2 = 0
   stime= 0
   etime= 0
   num3 = 0
   err1 = ""
   err2 = ""
   num4 = 0
   val1 = 0
   val2 = 0
   value = 0
   error =""
   valu_sel = 0
   err_sel = ""
   mloop= .t.
   fldname[1,1] = ""
   file1a = " "
   file2a = " "
   cnt = 2
   level[1] = 30
   DO WHILE cnt <= 12
      level[cnt] = 0
      cnt = cnt + 1
   ENDDO
   * Set up environment
   SET DEVELOPMENT on
   SET DELIMITERS  off
   SET SCOREBOARD  off
   SET ESCAPE      on
   SET CLOCK       off
   SET STATUS      off
   SET TALK        off
   SET EXACT       off
   SET BELL        off
   SET CATALOG     off

   * Define help key
   ON KEY LABEL F1 DO Helper

   * Set display characteristics - depends on hardware
   IF ISCOLOR()
      c_normal = "W+/B,W+/BG,B"
      c_pop    = "B+/W+,W+/RB+"
      red      = "R/W"
```

```
      blue     = "RB/W"
      lt_blue  = "W/BG"
      c_frame  = "W+/RB+"
      c_data = "W+/BG"
   ELSE
      STORE "W+/N,N/W" TO c_normal, c_pop
      STORE "N/W"      TO c_frame
      STORE "W+/N"     TO c_data
      STORE "W"        TO red, blue
      STORE "N/W"      TO lt_blue
   ENDIF
   * Define popup
   DO Main_def
RETURN

PROCEDURE Title
   CLEAR
   * Draw lines and box for menu with colors for effect
   @ 2,16 TO 5,63 COLOR N/N
   @ 1,15 CLEAR TO 4,62
   @ 1,15 TO 4,62 DOUBLE COLOR &blue.
   @ 1,15 FILL TO 4,62   COLOR &blue.
   SET COLOR TO &red.
   @ 2,33 SAY "CLIMATE DATA"
   @ 3,27 SAY "PROCESSING AND GRAPHING"
   SET COLOR TO &c_normal.
RETURN

PROCEDURE Main_def
   * Defines the main popup menu
   DEFINE POPUP MAINOUT FROM 7,11 TO 22,66 ;
     MESSAGE "Press first number of menu choice, or highlight and press <Enter>"
   DEFINE BAR  1 OF MAINOUT PROMPT "            ===== MAIN  MENU =====" SKIP
   DEFINE BAR  2 OF MAINOUT PROMPT "1 FOREST NURSING DEGREE DAYS"
   DEFINE BAR  3 OF MAINOUT PROMPT "2 CALCULATE HEATING/COOLING DAYS"
   DEFINE BAR  4 OF MAINOUT PROMPT "3 CALCULATE CORN HEAT UNITS"
   DEFINE BAR  5 OF MAINOUT PROMPT "4 CALCULATE DEGREE DAYS"
   DEFINE BAR  6 OF MAINOUT PROMPT "5 24 HOUR QUALITY CONTROL GRAPHS"
   DEFINE BAR  7 OF MAINOUT PROMPT "6 EXECUTE PC WINDROSE"
   DEFINE BAR  8 OF MAINOUT PROMPT "7 WINDCHILL & HUMIDEX CONVERSIONS" SKIP
   DEFINE BAR  9 OF MAINOUT PROMPT "8 SELECT DATA & INVOKE GRAPHING/STATISTICAL PRG"
   DEFINE BAR 10 OF MAINOUT PROMPT "9 SOIL TEMPERATURE PROFILES"
   DEFINE BAR 11 OF MAINOUT PROMPT "0 VERTICAL PROFILES"
   DEFINE BAR 12 OF MAINOUT PROMPT "E EDIT PROGRAM SET-UP FILES"
   DEFINE BAR 13 OF MAINOUT PROMPT "R RETURN TO ARCHIVE MAIN MENU"
   DEFINE BAR 14 OF MAINOUT PROMPT "Q QUIT TO DOS"
   ON SELECTION POPUP MAINOUT DO Main
RETURN

PROCEDURE Main
   * Execute case depending on user's choice
   CLEAR TYPEAHEAD
   ls=30
   DO CASE
      CASE BAR() =  2
           DO OUTFORM1
           DO PRODIC
           DO GDEG
           DO EXTPLOT1
      CASE BAR() =  3
           DO OUTFORM1
           DO PRODIC
           DO HEATCOOL
           DO EXTPLOT1
      CASE BAR() =  4
           DO OUTFORM1
           DO PRODIC
           DO CORNHEAT
           DO EXTPLOT1
      CASE BAR() =  5
           DO OUTFORM1
           DO PRODIC
           DO DEGDAYS
           DO EXTPLOT1
      CASE BAR() =  6
           DO EXTRACT1
           RUN GRAPH4
      CASE BAR() =  7
           DO EXTROSE
           RUN WINDROSE
      CASE BAR() =  9
           DO SELEC
           IF finished
             DO FILEPROC WITH "SELECT"
           ENDIF
           DO EXTPLOT
      CASE BAR() = 10
           DO INTERPOL
      CASE BAR() = 11
           DO VERTPROF
         RUN VERTGRAF
      CASE BAR() = 12
           DO TOEDIT
      CASE BAR() = 13 .OR. BAR() = 14
           IF BAR() = 13
              SET COLOR TO &C_NORMAL
              RETURN TO AES_SYS
           ELSE
              QUIT
           ENDIF
      ENDCASE
      CLEAR TYPEAHEAD
   RETURN TO AES_OUT
************************  END OF AES_OUT.PRG  ************************
```

```
PROCEDURE TOEDITSC
PUBLIC YR1,DY1,MO1
*************************************************************************
*    PROGRAM TO EDIT DATABASE FILES USED BY:                            *
*                          STA_CONT                                     *
*                                                                       *
*         WRITTEN AUGUST 1992  K. SINGH                                 *
*         LAST MODIFIED SEPT 10, 92   K. SINGH                          *
*************************************************************************
SELECT 1
SWIT6 = .F.
* FILE2A = FILE TO BE USED FOR EDITING. IN TOEDIT, A POPUP SELECTION
*          DETERMINES FILE2A
file2a="STA_CONT"
DO DISFIL
SET COLOR TO &c_normal
CLOSE DATABASES
RETURN


PROCEDURE DISFIL
*   SHOW CONTENTS OF FILES
DECLARE NOFIELD[20],FIELDNE[20]
* AREA 1 IS ALWAYS USED FOR THE FILE BEING EDITED
SELECT 1
USE &file2a
SET COLOR TO &c_normal
SET COLOR TO ,,G
CLEAR
* F10 IS THE CHANGE KEY
SET FUNCTION F10 TO ";"
* F9 IS THE DELETE KEY
SET FUNCTION F9 TO ";"
* F8 IS THE INSERT KEY
SET FUNCTION F8 TO ";"
TITLE="FILE IN USE: " + file2a
@ 0,1 SAY TITLE
@ 18,50 TO 24,79 COLOR &C_FRAME
@ 19,51 SAY "TO ENTER DATA PRESS    <F10>"
@ 20,51 SAY "TO DELETE RECORD PRESS <F9>"
@ 21,51 SAY "TO ADD RECORD PRESS    <F8>"
@ 22,51 SAY "pgdn SELECTS NEXT RECORD"
@ 23,51 SAY "pgup SELECTS PREVIOUS RECORD"
DO WHILE .NOT. EOF() .AND. .NOT. BOF()
  SWIT = .f. && SWITCH FOR GOING TO NEXT OR PREVIOUS RECORD
  SWIT1= .f. && SWITCH FOR END OF "FIELDS"
  SWIT3= .f. && USED FOR FLAGING AN INDEXED FIELD
  SWIT5= .f. && SKIP - 2 SWITCH
  SWIT7= .f. && DELETE SWITCH
  SWIT8= .f. && INSERT SWITCH
  RINT=1 && CURSOR POSITION - USED TO DETERMINE FIELD NUMBER
  SINT=1 && SAME AS RINT
  DO WHILE .NOT. SWIT
   DO WHILE .NOT. SWIT1
      IF LEN(FIELD(RINT))=0
        && NO MORE FIELDS -- QUIT
        SWIT1 =.t.
        CLEAR GETS
        LOOP
      ENDIF
      && DISPLAY DATA IN FILE ONE RECORD AT A TIME - ALL FIELDS
      @ RINT,1 SAY FIELD(RINT)
      FNAME=FIELD(RINT)
      NOFIELD[RINT]=&FNAME
      FIELDNE[RINT]=&FNAME
      && HIGHLIGHT FIELDS - FOR CONSISTANCY
      @ RINT,12 GET NOFIELD[RINT] COLOR &c_data
      RINT=RINT+1
   ENDDO
     && GET USER INPUT AT DESIRED FIELD
     @ SINT,12 GET NOFIELD[SINT] COLOR &c_data
     READ
     IRPOS=SINT
     i=LASTKEY()
     R1=RECCOUNT()
     R2=RECNO()
   IF i = 27
     && ESCAPE
     RETURN
   ENDIF
   IF i = 18
     && GET PREVIOUS RECORD
     SWIT5=.t.
     SWIT=.t.
     LOOP
   ENDIF
   IF i = 3
     && GET NEXT RECORD
     SWIT=.t.
   ENDIF
   IF i = 5 .OR. i = 19
     && MOVE MOVE UP
     IF SINT > 1
       SINT=SINT-1
     ELSE
       SINT=RINT-1
     ENDIF
     LOOP
   ENDIF
   IF i = -7
     IF R1-R2=0
      && EOF -- INSERT AFTER LAST RECORD
      INSERT BLANK
      SKIP-1
     ELSE
      && INSERT BEFORE CURRENT RECORD
      INSERT BEFORE BLANK
      SWIT8 = .t.
```

```
      ENDIF
      SWIT= .t.
    ENDIF
    IF i = -8
      && DELETE RECORD
      && CONFIRM DELETE
      DO D_MENU
      ACTIVATE POPUP D_CONFIRM
      IF SWIT7
        DELETE
        SWIT = .t.
      ENDIF
    ENDIF
    IF i = -9
      && CHANGE FIELD
      SWIT4 = .f.
      IRPOS=SINT
      DO POPSREL WITH IRPOS,SWIT3,SWIT4
      IF SWIT4
        ACTIVATE POPUP s_sermenu
      ENDIF
      SET COLOR TO &c_normal
      SET COLOR TO ,,G
      SELECT 1
      IF LASTKEY() = 27
        && ESCAPE -- DON'T CHANGE FIELD
        OLDFIELD=FIELD(SINT)
        REPLACE &OLDFIELD WITH FIELDNE[SINT]
        @ SINT,12 GET FIELDNE[SINT] COLOR &c_data
        CLEAR GETS
      ENDIF
      SWIT6 = .F.
      CLEAR GETS
    ENDIF
    NOFIELD[SINT]=FIELDNE[SINT]
    SINT=SINT+1
    IF SINT>RINT-1
      SINT=1
    ENDIF
  ENDDO
  SKIP
  IF SWIT8
    && INSERT AND REPOSTION TO PROPER RECORD NUMBER
    SKIP -1
  ENDIF
  IF SWIT7
    && PACK AND REPOSITION TO PROPER RECORD NUMBER
    PACK
    IRECPOS=RECCOUNT()-R2
    IF IRECPOS < 0
       GO BOTTOM
    ELSE
       GO R2
    ENDIF
  ENDIF
  IF SWIT5
    SKIP-2
  ENDIF
ENDDO
CLEAR GETS
RETURN

PROCEDURE POPSREL
PARAMETER IRPOS,SWIT3,SWIT4
SELECT 2
* TOEDSU DETERMINES WHICH FILES TO USE FOR THE EDITING FILE
* THE FIRST FIELD IN TOEDSU IS THE FILE NAME TO EDIT.  THE REST OF
* THE FIELDS DETERMINE WHICH FILE TO USE FOR THE POPUP.
* FOR EXAMPLE, IF FIELD(1) IN THE FILE "EXTDAT" IS THE SENSOR FIELD,
*  ENTER "EXTDAT" IN THE FIRST COLUMN AND "SENSOR" IN THE SECOND COLUMN.
* IF A FIELD CONTAINS AN INDEX, SPECIFY "INDEX" + INDEX NAME FOLLOWING IT
* (ie SENSOR INDEX SEN_MAKE).
* FOR A NON-RELATIONAL FIELD, ENTER A "DESCRIPTION" IN THE CORRESPONDING
* TOEDSU FIELD.
* TO PREVENT USER INPUT IN A SPECIFIED FIELD ENTER A "NOEDIT" FOLLOWING IT
* (ie SENSOR NOEDIT)
USE TOEDSU
LOCATE FOR SETUPFILE = file2a
IF EOF()
   @ 1,35 SAY "SET-UP FILE NOT CONFIGURED - PLEASE CHECK"
   RETURN
ENDIF
TOUSE=FIELD(IRPOS+1)
TOUSE=&TOUSE
T1=LTRIM(RTRIM(TOUSE))
IF LIKE("*NOEDIT*",T1)
   && PROTECTED FIELD
   TOUSE=SUBSTR(T1,1,LEN(T1)-6)
   SWIT6=.T. && USED TO DETERMINE A PROTECTED FIELD
   SELECT 1
   STATE=FIELD(IRPOS)
   IF LEN(LTRIM(RTRIM(&STATE)))=0
       && NO VALUE - USER INSERTED A NEW RECORD - ALLOW EDITING
       &&            OF PROTECTED FIELD
       SWIT6=.F.
   ENDIF
   SELECT 2
ENDIF
IF T1="DESCRIPTION"
  && NO POPUP -- RELATION DOES NOT EXIST -- EDIT MANUALY
  SET INTENSITY OFF
  DONE = .f.
  DO WHILE .NOT. DONE
    SELECT 1
    MODFIELD=FIELD(IRPOS)
    @ SINT,12 GET &MODFIELD COLOR W+,N
    READ
    ii=LASTKEY()
```

```
    IF ii=13
      && ENTER PRESSED -- REPLACE OLD VALUE WITH NEW VALUE
      NOFIELD[IRPOS]=&MODFIELD
      @ SINT,12 SAY NOFIELD[IRPOS] COLOR &c_data
      DONE = .t.
    ENDIF
    IF ii = 27
      &&  ESCAPE
      SET INTENSITY ON
      RETURN
    ENDIF
  ENDDO
  SET INTENSITY ON
  RETURN
ENDIF
SET COLOR TO &c_pop
SET COLOR OF HIGHLIGHT TO &lt_blue
SWIT4=.t.
USE &TOUSE
&&  TOUSE = RELATIONAL FILE USED FOR FIELD
SWIT3 = .f.
IF LIKE("*INDEX*",T1)
  && IF FILE INDEX SET SWITCH ON AND USE ALTERNATE FIELD
  SWIT3= .t.
ENDIF
   cnt = 0
   max = 18
   CHECK=" "
   DO WHILE .NOT. EOF()
      cnt = cnt + 1
      IF cnt > 9
          pos = CHR(55+cnt)
      ELSE
          pos = STR(cnt,1,0)
      ENDIF
      IA=3 && FIRST COLUMN USED FOR POPUP MENU
      IB=4 && SECOND COLUMN USED FOR POPUP MENU
      IF T1="OBSNUM"
          IA = 2
          IB = 3
      ENDIF
      IF T1="DERIVED"
          IA = 4
          IB = 120 && NONEXISTING FIELD
      ENDIF
      IF (T1="SENSOR" .AND. SWIT3=.F.) .OR. T1="ARR_DEF" .OR. T1="HEIGHT"
          IB = 120 && NONEXISTING FIELD
      ENDIF
      && FLx IS USED FOR DISPLAYING APPROPRIATE FIELDS IN POPUP
      FL1=FIELD(1)
      FL2=FIELD(2)
      FL3=FIELD(IA)
      FL4=FIELD(IB)
      IF LEN(FL4) = 0
          DUMMY=" "
          FL4="DUMMY"
      ENDIF
      CHECK=" " + LTRIM(RTRIM(&FL3)) + " " + LTRIM(RTRIM(&FL4))
      IF cnt > 1
          IF CHECK=CHECK1
            SKIP
          ENDIF
      ENDIF
      IF .NOT. SWIT3
          fldname[cnt,1] = &FL1 + " " + LTRIM(RTRIM(&FL3)) + " " + LTRIM(RTRIM(&FL4))
      ELSE
          fldname[cnt,1] = pos + " " + LTRIM(RTRIM(&FL3)) + " " + LTRIM(RTRIM(&FL4))
      ENDIF
      fldname[cnt,2] = &FL1
      fldname[cnt,3] = &FL2
      IF LEN(fldname[cnt,1]) > max
          max = LEN(fldname[cnt,1])
      ENDIF
      SKIP
      CHECK1=SUBSTR(fldname[cnt,1],2,LEN(fldname[cnt,1])-1)
   ENDDO
   &&CLOSE DATABASES
   cnt = cnt + 1
   fldname[cnt,1] = "X EXIT"
   fldname[cnt,2] = "0"
   fldname[cnt,3] = "0"
   tp = 0
   bt = tp + cnt + 2
   IF bt > 24
      bt = 24
   ENDIF
   level[4] = level[3] + 2
   ls1 = level[4]
   ls1=40
   rs = ls1 + max + 1
   IF rs > 72
      rs1 = rs - 72
      ls1 = ls1 - rs1
      IF ls1 < 1
          ls1 = 1
      ENDIF
      rs = 72
   ENDIF
   ls = ls1
   level[4] = ls
   * Defines the s_senmenu popup menu
   IF SWIT6
     DEFINE POPUP s_senmenu FROM tp,ls TO bt,rs ;
       MESSAGE "PROTECTED FIELD -- PRESS ENTER OR ESC TO CONTINUE"
   ELSE
     DEFINE POPUP s_senmenu FROM tp,ls TO bt,rs ;
       MESSAGE "Press first number of menu choice, or highlight and press <Enter>"
   ENDIF
```

```
   DEFINE BAR  1 OF s_senmenu PROMPT "CODE SELECT MENU" SKIP
   n1 = 1
   DO WHILE n1 <= cnt
      n1 = n1 + 1
      DEFINE BAR n1 OF s_senmenu PROMPT [!dname[n1-1,1]
   ENDDO
   ON SELECTION POPUP s_senmenu DO s_sen1 WITH SWIT3,IRPOS
RETURN

PROCEDURE s_sen1
PARAMETER SWIT3,IRPOS
   IF BAR() > cnt .OR. SWIT6 = .T.
      SWIT6 = .F.
      mloop = .f.
      RETURN TO DISFIL
   ENDIF

   IF .NOT. SWIT3
      REPVAL = [!dname[BAR()-1,2]
   ELSE
      REPVAL = [!dname[BAR()-1,3]
   ENDIF
   SELECT 1
   TOREPLAC=FIELD(IRPOS)
   REPLACE &TOREPLAC WITH REPVAL
   @ IRPOS,12 SAY REPVAL COLOR &s_data
   NOFIELD[IRPOS]=REPVAL
   FIELDNE[IRPOS]=REPVAL
RETURN TO DISFIL


PROCEDURE D_MENU
*   DEFINES PROGRAM SELECTION MENU
DEFINE POPUP D_CONFIRM FROM 13,50 TO 17,79 ;
  MESSAGE "Press first number of menu choice, or highlight and <ENTER>"

DEFINE BAR  1 OF D_CONFIRM PROMPT "      CONFIRM DELETE" SKIP
DEFINE BAR  2 OF D_CONFIRM PROMPT "N DO NOT DELETE RECORD"
DEFINE BAR  3 OF D_CONFIRM PROMPT "Y DELETE RECORD"
ON SELECTION POPUP D_CONFIRM DO S_DEL
RETURN

PROCEDURE S_DEL
DO CASE
  CASE BAR() =  2
     && DO NOT DELETE
  CASE BAR() =  3
     && DELETE
     SWIT7 = .T.
ENDCASE
RETURN TO DISFIL
```

```
PROCEDURE TOEDIT
PUBLIC YR1,DY1,MO1
*************************************************************************
*    PROGRAM TO EDIT DATABASE FILES USED BY:                            *
*                    1) EXTRACT,EXTRACT1                                *
*                    2) GDEG,CORNHEAT,HEATCOOL,DEGDAYS,FORMS            *
*                    3) HUMCHILL                                        *
*                    4) EXTROSE                                         *
*                    5) INTERPOL(SOIL TEMPERATURE PORFILES)             *
*                                                                       *
*         WRITTEN AUGUST 1992  K. SINGH                                 *
*         LAST MODIFIED AUG 31,92     K. SINGH                          *
*************************************************************************
SELECT 1
* FOR MORE DOCUMENTATION SEE TOEDITSC.PRG
* SELECT WHICH PROGRAM TO EDIT
SWIT6 = .F.
DO S_PROGR
ACTIVATE POPUP S_PRGMENU
IF LASTKEY() = 27 .OR. LASTKEY() = 19
  SET COLOR TO &c_normal
  RETURN
ENDIF
DO DISFIL
SET COLOR TO &c_normal
CLOSE DATABASES
RETURN


PROCEDURE S_PROGR
*   DEFINES PROGRAM SELECTION MENU
DEFINE POPUP S_PRGMENU FROM 10,10 TO 18,40 ;
  MESSAGE "Press first number of menu choice, or highlight and <ENTER>"

DEFINE BAR  1 OF S_PRGMENU PROMPT "FILE EDIT MENU" SKIP
DEFINE BAR  2 OF S_PRGMENU PROMPT "1 GRAPH4 GRAPHICS SETUP"
DEFINE BAR  3 OF S_PRGMENU PROMPT "2 WINDROSE SETUP"
DEFINE BAR  4 OF S_PRGMENU PROMPT "3 FORMS SETUP"
DEFINE BAR  5 OF S_PRGMENU PROMPT "4 VERTICAL PROFILE SETUP"
DEFINE BAR  6 OF S_PRGMENU PROMPT "5 SOIL TEMP PROFILE SETUP"
ON SELECTION POPUP S_PRGMENU DO S_PRGCAS
RETURN

PROCEDURE S_PRGCAS
DO CASE
  CASE BAR() =  2
     file2a="EXTDAT"
  CASE BAR() =  3
     file2a="ROSESU"
  CASE BAR() =  4
     file2a="FORFORMS"
  CASE BAR() =  5
     file2a="VERTSU"
  CASE BAR() =  6
     file2a="INTERPSU"
ENDCASE
RETURN TO TOEDIT

PROCEDURE DISFIL
*   SHOW CONTENTS OF FILES
DECLARE NOFIELD[20],FIELDNE[20]
SELECT 1
USE &file2a
SET COLOR TO &c_normal
SET COLOR TO ,,G
CLEAR
SET FUNCTION F10 TO ";"
SET FUNCTION F9 TO ";"
TITLE="FILE IN USE: " + file2a
@ 0,1 SAY TITLE
@ 20,50 TO 24,79 COLOR &C_FRAME
@ 21,51 SAY "TO ENTER DATA PRESS <F10>"
@ 22,51 SAY "pgdn SELECTS NEXT RECORD"
@ 23,51 SAY "pgup SELECTS PREVIOUS RECORD"
DO WHILE .NOT. EOF() .AND. .NOT. BOF()
  SWIT = .f.
  SWIT1= .f.
  SWIT3= .f.
  SWIT5= .f.
  RINT=1
  SINT=1
  DO WHILE..NOT. SWIT
   DO WHILE .NOT. SWIT1
     IF LEN(FIELD(RINT))=0
       && NO MORE FIELDS -- QUIT
       SWIT1 =.t.
       CLEAR GETS
       LOOP
     ENDIF
     @ RINT,1 SAY FIELD(RINT)
     FNAME=FIELD(RINT)
     NOFIELD[RINT]=&FNAME
     FIELDNE[RINT]=&FNAME
     @ RINT,12 GET NOFIELD[RINT] COLOR &c_data
     RINT=RINT+1
   ENDDO
     @ SINT,12 GET NOFIELD[SINT] COLOR &c_data
     READ
     IRIOS=SINT
     i=LASTKEY()
   IF i = 27
     && ESCAPE
     RETURN
   ENDIF
   IF i = 18
     && GET PREVIOUS RECORD
     SWIT5=.t.
     SWIT=.t.
```

```
      LOOP
    ENDIF
    IF i = 3
      && GET NEXT RECORD
      SWIT=.t.
    ENDIF
    IF i = 5 .OR. i = 19
      && MOVE MOVE UP
      IF SINT > 1
        SINT=SINT-1
      ELSE
        SINT=RINT-1
      ENDIF
      LOOP
    ENDIF
    IF i = -9
      && CHANGE FIELD
      SWIT4 = .f.
      IRPOS=SINT
      DO POPSREL WITH IRPOS,SWIT3,SWIT4
      IF SWIT4
        ACTIVATE POPUP s_semmenu
      ENDIF
      SET COLOR TO &c_normal
      SET COLOR TO ,,G
      SELECT 1
      IF LASTKEY() = 27
        && ESCAPE -- DON'T CHANGE FIELD
        OLDFIELD=FIELD(SINT)
        REPLACE &OLDFIELD WITH FIELDNE[SINT]
        @ SINT,12 GET FIELDNE[SINT] COLOR &c_data
        CLEAR GETS
      ENDIF
      SWIT5 = .F.
      CLEAR GETS
    ENDIF
    NOFIELD[SINT]=FIELDNE[SINT]
    SINT=SINT+1
    IF SINT>RINT-1
      SINT=1
    ENDIF
  ENDDO
  SKIP
  IF SWIT5
    SKIP-2
  ENDIF
ENDDO
CLEAR GETS
RETURN

PROCEDURE POPSREL
PARAMETER IRPOS,SWIT3,SWIT4
SELECT 2
USE TOEDSU
LOCATE FOR SETUPFILE = file2a
TOUSE=FIELD(IRPOS+1)
TOUSE=&TOUSE
T1=LTRIM(RTRIM(TOUSE))
IF LIKE("*NOEDIT**",T1)
    TOUSE=SUBSTR(T1,1,LEN(T1)-6)
    SWIT6=.T.
ENDIF
IF T1="DESCRIPTION"
  && NO POPUP -- RELATION DOES NOT EXIST -- EDIT MANUALY
  SET INTENSITY OFF
  DONE = .f.
  DO WHILE .NOT. DONE
    SELECT 1
    MODFIELD=FIELD(IRPOS)
    @ SINT,12 GET &MODFIELD COLOR W+/N
    READ
    ii=LASTKEY()
    IF ii=13
      && ENTER PRESSED -- REPLACE OLD VALUE WITH NEW VALUE
      NOFIELD[IRPOS]=&MODFIELD
      @ SINT,12 SAY NOFIELD[IRPOS] COLOR &c_data
      DONE = .t.
    ENDIF
    IF ii = 27
      &&  ESCAPE
      SET INTENSITY ON
      RETURN
    ENDIF
  ENDDO
  SET INTENSITY ON
  RETURN
ENDIF
SET COLOR TO &c_pop
SET COLOR OF HIGHLIGHT TO &lr_blue
SWIT4=.t.
USE &TOUSE
&&  TOUSE = RELATIONAL FILE USED FOR FIELD
SWIT3 = .f.
IF LIKE("*INDEX**",T1)
  && IF FILE INDEX SET SWITCH ON AND USE ALTERNATE FIELD
  SWIT3= .t.
ENDIF
   cnt = 0
   max = 18
   CHECK=" "
   DO WHILE .NOT. EOF()
      cnt = cnt + 1
      IF cnt > 9
          pos = CHR(55+cnt)
      ELSE
          pos = STR(cnt,1,0)
      ENDIF
      IA=1
```

Date Due

BRODART, INC Cat. No. 23 233 Printed in U.S.A