UNITED STATES DEPARTMENT OF INTERIOR

GEOLOGICAL SURVEY

GARNETAB

A Computer Program to Create an Absentee File for Executing GARNET

(Graphic Analysis of Resources Using Numeric Evaluation Techniques).

By

William A. Scott (1)
&
Sharon S. Crowley (1)

Open-File Report 83-617

(1) Reston, Virginia

1983

Table of Contents

INTRODUCTION

GARNETAB, an interactive program in the National Coal Resources Data System (NCRDS), creates a Multics absentee file to operate GARNET (Graphic Analysis of Resources Using Numeric Evaluation Techniques).

PURPOSE

By using GARNETAB, the user can run the GARNET program in the absentee mode, during hours of low computer cost or at another time up to the user's descrestion. GARNETAB creates a file containing GARNET commands and input, which can be run after working hours.

GARNETAB is designed for experienced GARNET users.

USING GARNETAB IN THE MULTICS ENVIRONMENT

As in GARNET, one must be a registered Multics user in order to access and execute GARNETAB. The user may link to GARNETAB by entering the following command:

```
lk >udd>ORA>library>garnetab
```

If the user has NCRDS search rules, the GARNETAB program will be found automatically.

SESSION EXAMPLE

The sample run which follows illustrates the dialogue between user and the GARNETAB (GARNET absentee) program:

Computer prompts are in uppercase, user input is in lowercase.

Supplemental information from the authors is in parentheses.

(To begin, the user enters 'garnetab')

```
garnetab

'garnetab'   CREATES A MULTICS  ABSENTEE INPUT FILE
CONTAINING GARNET COMMANDS AND FUNCTIONS THAT WOULD
BE ENTERED IF  GARNET WERE EXECUTED  INTERACTIVELY.

YOU MAY CHANGE WORKING DIRECTORIES
AT THIS TIME (cwd >udd>NCRDS>NEWdir ...),
OTHERWISE PRESS C/R:  cwd >udd>NCRDS>NEWper


ENTER NAME FOR YOUR ABSENTEE REQUEST-
   The  garnetab  program  adds  all  suffixes:  sampl


ENTER NAME FOR 'TEKTRONIX' OUTPUT FILE:  tekout


FOR A LIST OF INSTRUCTIONS, ENTER THE COMMAND 'help'
```

```
ENTER COMMAND:  help

THE AVAILABLE ABSENTEE COMMANDS ARE:

HELP:       PRINTS THIS LIST.

GRID:       CREATES A GRIDDED FILE WHICH IS NEEDED TO PRODUCE CONTOUR PLOTS.

CONTOUR:    PRODUCES A CONTOUR MAP FROM A GRIDDED DATA FILE.

LOCATE:     'REFER TO THE CONTOUR COMMAND.'

WINDOW:     ALLOWS THE USER TO SELECT A NEW MAP WINDOW.
            USING KEYED-IN COORDINATES ONLY)

UNWINDOW:   RESETS THE WINDOW TO THE ORIGINAL MAP FRAME.

BOUNDARY:   PERMITS THE CREATION OF BOUNDED REGIONS.
            ALSO PERMITS THE LOGICAL COMBINATIONS OF BOUNDARY LINES.

RESOURCE:   COMPUTES RESOURCE VOLUMES AND TONNAGES AND PLOTS A RESOURCE MAP
            FOR GIVEN RELIABILITY CATEGORY DISTANCES.

QUIT:       TERMINATES GARNET AND RETURNS TO MULTICS COMMAND LEVEL.

            NOTE THAT ANY OF THESE COMMANDS MAY BE EXECUTED BY ENTERING
            ONLY THE FIRST FOUR CHARACTERS OF THE COMMAND.
```

```
ENTER COMMAND: grid

DO YOU WISH TO WEIGHT THE NEW SET OF GRID VALUES WITH THE VALUES
FROM AN OLD GRID? yes

NAME OF GRIDDED INPUT FILE: testg1 (if above is no, this line is skipped)

NAME OF OBSERVED POINT FILE: test01

ENTER FRACTION OF MAP TO BE PLOTTED OUTSIDE OF RECTANGULAR BOUNDARY: 0.1

GRID SIZE IN MILES: .2

IS THE DATA DENSE AND DISTRIBUTED UNIFORMLY? yes

ARE NEGATIVE Z-VALUES ACCEPTABLE? no

A NEW GRIDDED DATA FILE HAS BEEN CREATED.
ENTER FILE NAME TO STORE GRID VALUES:  savegv
```

```
ENTER COMMAND: contour                                                  7

NAME OF OBSERVED POINT FILE: test01

ENTER A '1' IF YOU WANT ONLY THE Z-VALUES PLOTTED,
        '2' IF YOU WANT ONLY THE POINTIDS PLOTTED,
        '3' IF YOU WANT BOTH PLOTTED.

ENTER NUMBER: 2

NAME OF GRIDDED INPUT FILE: gfile

NAME OF GRIDDED INPUT FILE: testg1

CONTOUR INTERVAL: 14

FREQUENCY OF BOLD LINES: 1

FOR CONTOUR SMOOTHING, ENTER A '1':

TO SPECIFY GRAPHICS DEVICE, ENTER '0' FOR TEKTRONIX ONLY,
                            ENTER '1' FOR TEKTRONIX AND PLOTTER,
                         OR ENTER '2' FOR PLOTTER ONLY.

GRAPHICS DEVICE (0,1,2): 1   (or '2' will give the following prompts)


BOUNDARY FILE NAME:    bfile

BOUNDARY FILE NAME  2: none
(If selected graphics device is '0', next prompt - 'ENTER COMMAND:')

ENTER X AXIS TITLE (< 31 chars): X TITLE

ENTER Y AXIS TITLE (< 31 chars): Y TITLE

NAME OF PLOT FILE: plot1

ENTER MAP SCALE: 24000

TO SPECIFY PLOTTING DEVICE, ENTER '1' FOR CALCOMP,
                            ENTER '2' FOR VERSATEC

ENTER PLOTTING DEVICE: 2
```

```
ENTER SYMBOL SIZE FOR LINE AND POINT LABELS (DECIMAL INCHES): .08


ENTER COMMAND: locate

NAME OF OBSERVED POINT FILE: test01
(If 'none' is entered, next prompt - 'NAME OF GRIDDED INPUT FILE:')

ENTER A '1' IF YOU WANT ONLY THE Z-VALUES PLOTTED,
        '2' IF YOU WANT ONLY THE POINTIDS PLOTTED,
        '3' IF YOU WANT BOTH PLOTTED.

ENTER NUMBER: 2

ENTER FRACTION OF MAP TO BE PLOTTED OUTSIDE OF RECTANGULAR BOUNDARY: 0.

TO SPECIFY GRAPHICS DEVICE, ENTER '0' FOR TEKTRONIX ONLY,
                            ENTER '1' FOR TEKTRONIX AND PLOTTER,
                         OR ENTER '2' FOR PLOTTER ONLY.

GRAPHICS DEVICE (0,1,2): 1

BOUNDARY FILE NAME:    boundf

BOUNDARY FILE NAME  2: none
(If selected graphics device is '0', next prompt - 'ENTER COMMAND:')

ENTER X AXIS TITLE (< 31 chars): X TITLE

ENTER Y AXIS TITLE (< 31 chars): Y TITLE

NAME OF PLOT FILE: plotfl

ENTER MAP SCALE: 24000

TO SPECIFY PLOTTING DEVICE, ENTER '1' FOR CALCOMP,
                            ENTER '2' FOR VERSATEC

ENTER PLOTTING DEVICE: 2

ENTER SYMBOL SIZE FOR LINE AND POINT LABELS (DECIMAL INCHES): .08
```

```
ENTER COMMAND: window

NAME OF GRIDDED INPUT FILE: gridin

ENTER A '2' FOR KEYBOARD INPUT,
        '3' FOR FILE INPUT.

(A '2' entry will cause the next prompt to be WEST)

ENTER NUMBER: 3

ENTER WINDOW FILE NAME: windnm

ENTER WINDOW NAME: wind1

WEST : 82 20 00

EAST : 82 15 00

NORTH: 36 56 30

SOUTH: 36 52 30

DO YOU WANT TO SAVE THIS WINDOW: yes
(If "no", ENTER COMMAND appears)

ENTER OUTPUT WINDOW FILE NAME: newwind

ENTER WINDOW NAME: wname



ENTER COMMAND: unwindow
```

```
ENTER COMMAND: boundary

AVAILABLE ABSENTEE BOUNDARY SUB-COMMANDS

      combine
      reverse
      window
      unwindow
      quit

FOR A LIST OF INSTRUCTIONS ENTER THE COMMAND 'help'
```

```
ENTER BOUNDARY COMMAND: combine

DEFINE BOUNDARY FRAME: testg1

DO YOU NEED INSTRUCTIONS? yes

Combine Instructions

Enter boundary combinations in the following format:

   outputfile=boundary1.logic.boundary2

   logic can be: and
                 or
                 not

Example:

   1.  comb1=outcrop1.and.county
   2.  comb2=comb1.and.overbrdn
   3.  comb3=comb2.or.iso42
   4.  comb4=comb3.not.stripped
   5.  comb5=outcrop2.and.county
   6.  carriage return

   Output files can be any name not exceeding 8 characters.

   You may combine files created in a previous step.

   Only the final combination is displayed and shaded.  All
   boundary files created in this command are saved, and you
   may display any of them using the 'display' command.


         1  bound12=bound1.and.bound2

         2  bound3=bound3.or.bound4

         3  (c/r pressed)
```

```
ENTER BOUNDARY COMMAND: reverse

DEFINE BOUNDARY FRAME: testg1

BOUNDARY FILE NAME: greenco
```

```
ENTER BOUNDARY COMMAND: window

NAME OF GRIDDED INPUT FILE: gfile1

ENTER A '2' FOR KEYBOARD INPUT,
        '3' FOR FILE INPUT.

ENTER NUMBER: 2

WEST : 72 30 00

EAST : 72 22 30

SOUTH: 40 00 00

NORTH: 40 07 30

DO YOU WANT TO SAVE THIS WINDOW: yes
(If 'no' is entered, next prompt is ENTER COMMAND:)

ENTER OUTPUT WINDOW FILE NAME: windout

ENTER WINDOW NAME: windnm

ENTER BOUNDARY COMMAND: unwindow

ENTER BOUNDARY COMMAND: quit
```

```
ENTER COMMAND: resource                                              14

(There are several different resource options in this session.)

AT EACH PROMPT, ENTER THE RELIABILITY RADIUS IN MILES.
ENTER THE SMALLEST RADIUS AT THE PROMPT AND CONTINUE TO ENTER THE RADII
IN ASCENDING ORDER.  AFTER THE LAST RADIUS HAS BEEN ENTERED, ENTER
AN '0' AT THE NEXT PROMPT TO TERMINATE THE INPUT PROCEDURE.
IF YOU ENTER A '0' FOR THE FIRST RADIUS THEN THE U.S.G.S.
STANDARD RADII OF 0.25 MILE, 0.75 MILE, AND 3.0 MILES WILL BE USED.

ENTER RADIUS FOR RELIABILITY CATEGORY          1.  0


NAME OF OBSERVED POINT FILE: test01


ENTER A '1' IF YOU WANT ONLY THE Z-VALUES PLOTTED,
        '2' IF YOU WANT ONLY THE POINTIDS PLOTTED,
        '3' IF YOU WANT BOTH PLOTTED.

ENTER NUMBER: 1


NAME OF GRIDDED INPUT FILE: testg1


DO YOU WISH TO PLOT A RESOURCE MAP? yes


ENTER MAP SCALE: 24000


DO YOU WISH TO CALCULATE VOLUMES? (yes OR no): yes


DO YOU WISH TO PLOT THE RESOURCE MAP WITH A CONTOUR SURFACE? (yes OR no): y

(If response is 'no', the resource map is not plotted.  You receive only-to


CONTOUR INTERVAL: 14.


FREQUENCY OF BOLD LINES: 1


FOR CONTOUR SMOOTHING, ENTER A '1': 1


TO SPECIFY GRAPHICS DEVICE, ENTER '0' FOR TEKTRONIX ONLY,
                           ENTER '1' FOR TEKTRONIX AND CALCOMP,
                        OR ENTER '2' FOR CALCOMP ONLY.

GRAPHICS DEVICE (0,1, or 2): 0
```

```
BOUNDARY FILE NAME:     bfile

BOUNDARY FILE NAME  2: none

DO YOU WANT TO SAVE THE RESOURCE TABLE IN A FILE? yes

ENTER RESOURCE TABLE FILE NAME: rtable

ENTER TONNAGE PER ACRE FOOT: 1800.

ENTER STATE: virginia

ENTER COUNTY: fairfax

ENTER QUAD: quad name

ENTER BED: bed name

ENTER TOWNSHIP: towns

ENTER N/S: ns

ENTER RANGE: range name

ENTER E/W: ew name

ENTER THICKNESS CATEGORY: 7

ENTER OVERBURDEN CATEGORY: 3
```

```
ENTER COMMAND: reso


AT EACH PROMPT, ENTER THE RELIABILITY RADIUS IN MILES.
ENTER THE SMALLEST RADIUS AT THE PROMPT AND CONTINUE TO ENTER THE RADII
IN ASCENDING ORDER.  AFTER THE LAST RADIUS HAS BEEN ENTERED, ENTER
AN '0' AT THE NEXT PROMPT TO TERMINATE THE INPUT PROCEDURE.
IF YOU ENTER A '0' FOR THE FIRST RADIUS THEN THE U.S.G.S.
STANDARD RADII OF 0.25 MILE, 0.75 MILE, AND 3.0 MILES WILL BE USED.

ENTER RADIUS FOR RELIABILITY CATEGORY          1.  0


NAME OF OBSERVED POINT FILE: test01


ENTER A '1' IF YOU WANT ONLY THE Z-VALUES PLOTTED,
        '2' IF YOU WANT ONLY THE POINTIDS PLOTTED,
        '3' IF YOU WANT BOTH PLOTTED.

ENTER NUMBER: 2


NAME OF GRIDDED INPUT FILE: testg1


DO YOU WISH TO PLOT A RESOURCE MAP? no


BOUNDARY FILE NAME:     boundf

BOUNDARY FILE NAME  2: none


DO YOU WANT TO SAVE THE RESOURCE TABLE IN A FILE? no


ENTER TITLE (LESS THAN 31 CH): TITLE NAME


ENTER TONNAGE PER ACRE FOOT: 1800.
```

```
ENTER COMMAND: reso                                                    17

AT EACH PROMPT, ENTER THE RELIABILITY RADIUS IN MILES.
ENTER THE SMALLEST RADIUS AT THE PROMPT AND CONTINUE TO ENTER THE RADII
IN ASCENDING ORDER.  AFTER THE LAST RADIUS HAS BEEN ENTERED, ENTER
AN '0' AT THE NEXT PROMPT TO TERMINATE THE INPUT PROCEDURE.
IF YOU ENTER A '0' FOR THE FIRST RADIUS THEN THE U.S.G.S.
STANDARD RADII OF 0.25 MILE, 0.75 MILE, AND 3.0 MILES WILL BE USED.

ENTER RADIUS FOR RELIABILITY CATEGORY          1.  0


NAME OF OBSERVED POINT FILE: test01


ENTER A '1' IF YOU WANT ONLY THE Z-VALUES PLOTTED,
        '2' IF YOU WANT ONLY THE POINTIDS PLOTTED,
        '3' IF YOU WANT BOTH PLOTTED.

ENTER NUMBER: 3


NAME OF GRIDDED INPUT FILE: testg1


DO YOU WISH TO PLOT A RESOURCE MAP? yes


ENTER MAP SCALE: 24000


DO YOU WISH TO CALCULATE VOLUMES? (yes OR no): yes


DO YOU WISH TO PLOT THE RESOURCE MAP WITH A CONTOUR SURFACE? (yes OR no): yes

(If response is 'no', the resource map is not plotted.
 You receive only tonnage figures.)



CONTOUR INTERVAL: 14.


FREQUENCY OF BOLD LINES: 24000


FOR CONTOUR SMOOTHING, ENTER A '1':


TO SPECIFY GRAPHICS DEVICE, ENTER '0' FOR TEKTRONIX ONLY,
                           ENTER '1' FOR TEKTRONIX AND CALCOMP,
                        OR ENTER '2' FOR CALCOMP ONLY.

GRAPHICS DEVICE (0,1, or 2): 1     (or 2)


BOUNDARY FILE NAME:     bfile
```

```
BOUNDARY FILE NAME  2: none

ENTER X AXIS TITLE (< 31 chars): X TITLE

ENTER Y AXIS TITLE (< 31 chars): Y TITLE

NAME OF PLOT FILE: pfile

TO SPECIFY PLOTTING DEVICE, ENTER '1' FOR CALCOMP,
                            ENTER '2' FOR VERSATEC

ENTER PLOTTING DEVICE: 2

ENTER SYMBOL SIZE FOR LINE AND POINT LABELS (DECIMAL INCHES): .08
```

```
ENTER COMMAND: quit

Like to add (dp sampl.absin sampl.absout) to request ?
(Enter  yes OR no): yes


sampl.absin

Your absentee file has now been created
with the suffix (.absin) added.
```

ENTERING THE GARNETAB ABSENTEE REQUEST

(User should submit the request when ready.)

example:

```
     ear sampl.absin -q 4
```

```
cwd >udd>NCRDS>NEWper
setup_tektronix_tcs -reset
setup_tektronix_tcs -fl -nm tekout
garnet

grid
y
testg1
test01
0.1
.2
y
n
savegv
cont
test01
2
testg1
14
1

1
bfile
none
X TITLE
Y TITLE
plot1
24000
2
.08
loca
test01
2
gfile
0.
1
boundf
none
X TITLE
Y TITLE
plotfl
24000
2
.08
wind
gridin
3
windnm
wind1
82 20 00
82 15 00
36 56 30
36 52 30
y
newwind
wname
```

```
unwi
boun

comb
testg1
no
bound12=bound1.and.bound2
bound3=bound3.or.bound4


reve
testg1
greenco

wind
gfile1
2
72 30 00
72 22 30
40 00 00
40 07 30
y
windout
windnm
unwi
quit
reso
0
test01
1
testg1
y
24000
y
y
14.
1
1
0
bfile
y
rtable
1800.
virginia
fairfax
quad name
bed name
towns
ns
range name
ew name
7
3

reso
0
test01
2
testg1
n
```

```
boundf
n
TITLE NAME
1800.


reso
0
test01
3
testg1
y
24000
y
y
14.
24000

1
bfile
none
X TITLE
Y TITLE
pfile
2
.08
quit
dp sampl.absin sampl.absout
gr -set -string r -date -time -inc_vcpu -total_vcpu
-inc_pf -total_pf -inc_mem_units -total_mem_units -inc_cost -total_cost
logout
```

GARNETAB SOURCE LISTING

```
 (stringrange):
   (subscriptrange):
garnetab:
   proc;
   dcl (abfl) file;
   dcl (usrin) file;
   dcl (sysin,sysprint) file;
   dcl rename entry (char(*),char(*));
   open file (usrin) environment (stringvalue) record input
   title ("record_stream_ user_input");
   open file (abfl) stream output;
   dcl delete entry options (variable);
   dcl dp entry options (variable);
   dcl quit cond;
   dcl endfile cond;
   on quit go to brk;
   dcl usrnm   char (24) varying init ("");
   dcl keyin   char (24) varying init ("");
   dcl calcv   char (24) varying init ("");
   dcl titlein char (36) varying init ("");
   dcl tkey    char (24) varying init ("");
   dcl cmmd    char (16) varying init ("");
   dcl tusrnm  char (24) varying init ("");
   dcl reqi    char (24) varying init ("");
   dcl reqo    char (24) varying init ("");
   dcl dpareq  char (64) varying init ("");
   dcl treqi   char (64) varying init ("");
   dcl gdvice  char (4)  varying init ("");
   dcl tcmmd   char (16) varying init ("");
   dcl cmmd4   char (8)  varying init ("");
   dcl pdvice  char (4)  varying init ("");
   dcl inum    char (4)  varying init ("");
   dcl setname char (40) varying init ("");
   usrnm,keyin,tkey,cmmd,tusrnm,reqi,reqo,dpareq,treqi,gdvice,tcmmd,cmmd4,titlein,calcv = "";
   pdvice, inum, setname = "";
   dcl (flag, dot, nxnum, bcflag, uwflag,bcount, bflag, wflag,incnum) fixed bin;
   flag, dot, nxnum, bcflag, uwflag,bcount, bflag, wflag,bcount,incnum = 0;
   put skip (1) edit ("'garnetab'   CREATES A MULTICS  ABSENTEE INPUT FILE") (a);
   put skip (1) edit ("CONTAINING GARNET COMMANDS AND FUNCTIONS THAT WOULD") (a);
   put skip (1) edit ("BE ENTERED IF  GARNET WERE EXECUTED  INTERACTIVELY.") (a);
   put skip (2) edit ("YOU MAY CHANGE WORKING DIRECTORIES") (a);
   put skip (1) edit ("AT THIS TIME (cwd ...),") (a);
   put skip edit ("OTHERWISE PRESS C/R: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to ent2;
   put file (abfl) edit (keyin) (a);
   put file (abfl) skip (1) edit ("") (a);
```

```
ent2:
   put skip (2) edit ("ENTER NAME FOR YOUR ABSENTEE REQUEST-") (a);
   put skip (1) edit ("   The  garnetab  program  adds  all  suffixes: ") (a);
   read file (usrin) into (usrnm);
   if usrnm = "" then go to ent2;
begin:
   put skip (2) edit ("ENTER NAME FOR 'TEKTRONIX' OUTPUT FILE: ") (a);
   read file (usrin) into (tusrnm);
   if tusrnm = "" then go to begin;
   put file (abfl) edit ("setup_tektronix_tcs -reset") (a);
   setname = "setup_tektronix_tcs -fl -nm " || tusrnm;
   put file (abfl) skip (1) edit (setname) (a);
   put file (abfl) skip (1) edit ("garnet") (a);
   put file (abfl) skip (1) edit ("") (a);
   put skip (2) edit ("FOR A LIST OF INSTRUCTIONS, ENTER THE COMMAND 'help'") (a);
   go to ecmmd;
clist:
   put skip (2) edit ("The available absentee commands are:") (a);
   put skip (2) edit ("help:      Prints this list.") (a);
   put skip (2) edit ("grid:      Creates a gridded file which is needed to produce contour plots.") (a);
   put skip (2) edit ("reference: Plot reference points in tektronix and calcomp plot files.") (a);
   put skip (2) edit ("contour:   Produces a contour map from a gridded data file.") (a);
   put skip (2) edit ("locate:    'refer to the contour command.'") (a);
   put skip (2) edit ("window:    Allows the user to select a new map window.") (a);
   put skip (1) edit ("           Using keyed-in coordinates only)") (a);
   put skip (2) edit ("unwindow:  Resets the window to the original map frame.") (a);
   put skip (2) edit ("boundary:  Permits the creation of bounded regions.") (a);
   put skip (1) edit ("           Also permits the logical combinations of boundary lines.") (a);
   put skip edit ("") (a);
   put skip edit ("resource:  Computes resource volumes and tonnages and plots a resource map for")(a);
   put skip (1) edit ("           given reliability category distances.") (a);
   put skip (2) edit ("quit:      Terminates Garnet and returns to Multics Command Level.") (a);
   put skip (2) edit ("           Note that any of these commands may be executed by entering") (a);
   put skip (1) edit ("           only the first four characters of the command.") (a);
   put skip (1) edit ("") (a);
ecmmd:
   cmmd = "";
   put skip (3) edit ("ENTER COMMAND: ") (a);
   read file (usrin) into (cmmd);
   if cmmd = "" then go to ecmmd;
   tcmmd = cmmd || " xxxx";
   cmmd4 = substr (tcmmd,1,4);
   cmmd = cmmd4;
   if cmmd = "help" then go to clist;
```

```
grdc:
   if cmmd ^= "grid" then go to refec;
   go to gridr;
refec:
   if cmmd ^= "refe" then go to bouc;
   go to refer;
bouc:
   if cmmd ^= "boun" then go to resc;
   go to boundr;
resc:
   if cmmd ^= "reso" then go to conc;
   go to resor;
conc:
   if cmmd ^= "cont" then go to locc;
   go to contr;
locc:
   if cmmd ^= "loca" then go to winc;
   go to locr;
winc:
   if cmmd ^= "wind" then go to unwind;
   go to windr;
unwind:
   if cmmd ^= "unwi" then go to qt;
   put file (abfl) skip (1) edit (cmmd) (a);
   if uwflag = 1 then go to ebcmmd;
   go to ecmmd;
qt:
   if cmmd = "quit" then go to qtr;
   put skip (2) edit (cmmd, " command not available.") (a);
   go to clist;
gridr:
   put file (abfl) skip (1) edit (cmmd) (a);
weighin:
   put skip edit ("") (a);
   put skip edit ("DO YOU WISH TO WEIGHT THE NEW SET OF GRID VALUES WITH THE VALUES FROM AN OLD GRID? ")(a
   read file (usrin) into (keyin);
   keyin = substr (keyin,1,1);
   if (keyin ^= "y" & keyin ^= "n") then go to weighin;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "n" then go to point;
   put skip (2) edit ("NAME OF GRIDDED INPUT FILE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
point:
   put skip (2) edit ("NAME OF OBSERVED POINT FILE: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to point;
   put file (abfl) skip (1) edit (keyin) (a);
```

```
frac:
   put skip (2) edit ("ENTER FRACTION OF MAP TO BE PLOTTED OUTSIDE OF RECTANGULAR BOUNDARY: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to frac;
   put file (abfl) skip (1) edit (keyin) (a);
gsize:
   put skip (2) edit ("GRID SIZE IN MILES: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to gsize;
   put file (abfl) skip (1) edit (keyin) (a);
ddense:
   put skip (2) edit ("IS THE DATA DENSE AND DISTRIBUTED UNIFORMLY? ") (a);
   read file (usrin) into (keyin);
   keyin = substr (keyin,1,1);
   if (keyin ^= "y" & keyin ^= "n") then go to ddense;
   put file (abfl) skip (1) edit (keyin) (a);
zval:
   put skip (2) edit ("ARE NEGATIVE Z-VALUES ACCEPTABLE? ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to zval;
   keyin = (substr (keyin,1,1));
   put file (abfl) skip (1) edit (keyin) (a);
gfile:
   put skip (2) edit ("A NEW GRIDDED DATA FILE HAS BEEN CREATED.") (a);
   put skip (2) edit ("ENTER FILE NAME TO STORE GRID VALUES:   ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to gfile;
   tkey = keyin || "*";
   flag = search (substr (tkey,1), "*");
   if flag ^>9 then go to goodln;
   put skip (2) edit ("Must be <9 chars.") (a);
   go to gfile;
goodln:
   put file (abfl) skip (1) edit (keyin) (a);
   go to ecmmd;
refer:
   put file (abfl) skip (1) edit (cmmd) (a);
   put skip (2) edit ("NAME OF GRIDDED INPUT FILE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
eref:
   put skip (2) edit ("ENTER REFERENCE POINT FILE NAME: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to eref;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin ^= "input" then go to ecmmd;
```

```
eltln:
   put skip (2) edit ("ENTER LAT AND LONG IN FORMAT:  ddd mm ss") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to eltln;
   put file (abfl) skip (1) edit (keyin) (a);
   incnum = 1;
refpt:
   put skip (2) edit ("REFERENCE POINT LATITUDE  ", (substr(incnum,9,2)), ": ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "" then go to savref;
rplong:
   put skip (1) edit ("REFERENCE POINT LONGITUDE ", (substr(incnum,9,2)), ": ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "" then go to rplong;
   incnum = incnum + 1;
   go to refpt;
savref:
   put skip (2) edit ("TO SAVE REFERENCE POINTS ENTER FILE NAME: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   go to ecmmd;
boundr:
   bcflag = 0;
   put file (abfl) skip (1) edit (cmmd) (a);
bclist:
   put skip (2) edit ("AVAILABLE ABSENTEE BOUNDARY SUB-COMMANDS") (a);
goodbc:
   put skip (2) edit ("     combine") (a);
   put skip (1) edit ("     reverse") (a);
   put skip (1) edit ("     window") (a);
   put skip (1) edit ("     unwindow") (a);
   put skip (1) edit ("     quit") (a);
   put file (abfl) skip (1) edit ("") (a);
   bcflag = 1;
   cmmd = "";
   put skip (2) edit ("FOR A LIST OF INSTRUCTIONS ENTER THE COMMAND 'help'") (a);
ebcmmd:
   put skip (2) edit ("ENTER BOUNDARY COMMAND: ") (a);
   read file (usrin) into (cmmd);
   if cmmd = "" then go to ebcmmd;
   tcmmd = cmmd || " xxxx";
   cmmd4 = substr (tcmmd,1,4);
   cmmd = cmmd4;
   if cmmd = "help" then go to bclist;
```

```
bcombc:
   if cmmd ^= "comb" then go to brevc;
   go to bcombr;
brevc:
   if cmmd ^= "reve" then go to bwinc;
   go to brevr;
bwinc:
   if cmmd ^= "wind" then go to bunwc;
   bflag = 1;
   go to windr;
bunwc:
   if cmmd ^= "unwi" then go to bqt;
   go to unwind;
bqt:
   if cmmd = "quit" then go to bqtr;
   put skip (2) edit (cmmd, "BOUNDARY COMMAND NOT AVAILABLE.") (a);
   go to bclist;
bcombr:
   put file (abfl) skip (1) edit (cmmd) (a);
   put skip (2) edit ("DEFINE BOUNDARY FRAME: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
instr:
   put skip (2) edit ("DO YOU NEED INSTRUCTIONS? ") (a);
   read file (usrin) into (keyin);
   keyin = substr (keyin,1,1);
   if (keyin ^= "y" & keyin ^= "n") then go to instr;
   put file (abfl) skip (1) edit ("no") (a);
   if keyin = "n" then go to noinst;
   put skip (2) edit ("Combine Instructions") (a);
   put skip (3) edit ("Enter boundary combinations in the following format:") (a);
   put skip (2) edit ("   outputfile=boundary1.logic.boundary2") (a);
   put skip (2) edit ("   logic can be:  and") (a);
   put skip (1) edit ("                  or") (a);
   put skip (1) edit ("                  not") (a);
   put skip (2) edit ("Example:") (a);
   put skip (2) edit ("   1.  comb1=outcrop1.and.county") (a);
   put skip (1) edit ("   2.  comb2=comb1.and.overbrdn") (a);
   put skip (1) edit ("   3.  comb3=comb2.or.iso42") (a);
   put skip (1) edit ("   4.  comb4=comb3.not.stripped") (a);
   put skip (1) edit ("   5.  comb5=outcrop2.and.county") (a);
   put skip (1) edit ("   6.  carriage return") (a);
   put skip (2) edit ("   Output files can be any name not exceeding 8 characters.") (a);
   put skip (2) edit ("   You may combine files created in a previous step.") (a);
   put skip (2) edit ("   Only the final combination is displayed and shaded.   All") (a);
   put skip (1) edit ("   boundary files created in this command are saved, and you") (a);
   put skip (1) edit ("   may display any of them using the 'display' command.") (a);
noinst:
   bcount = 0;
   put skip (2) edit ("") (a);
```

```
nxcnt:
   bcount = bcount + 1;
   put skip (1) edit (bcount, "  ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
   if titlein ^= "" then go to nxcnt;
   put file (abfl) skip (1) edit ("") (a);
   go to ebcmmd;
brevr:
   put file (abfl) skip (1) edit (cmmd) (a);
dboundf:
   put skip (2) edit ("ENTER GRID FILE NAME: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to dboundf;
   put file (abfl) skip (1) edit (keyin) (a);
bbfile:
   put skip (2) edit ("BOUNDARY FILE NAME:    ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to bbfile;
   put file (abfl) skip (1) edit (keyin) (a);
   put file (abfl) skip (1) edit ("") (a);
savbf:
   put skip (2) edit ("DO YOU WISH TO SAVE THIS") (a);
   put skip (1) edit ("BOUNDARY FILE? ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to savbf;
   keyin = (substr(keyin,1,1));
   if (keyin ^= "y" & keyin ^= "n") then go to savbf;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "n" then go to ebcmmd;
outbf:
   put skip (2) edit ("NAME OF OUTPUT") (a);
   put skip (1) edit ("BOUNDARY FILE: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to outbf;
   put file (abfl) skip (1) edit (keyin) (a);
   go to ebcmmd;
bqtr:
   put file (abfl) skip (1) edit (cmmd) (a);
   go to ecmmd;
```

```
resor:
   put file (abfl) skip (1) edit (cmmd) (a);
   put skip (2) edit ("AT EACH PROMPT, ENTER THE RELIABILITY RADIUS IN MILES.") (a);
   put skip edit ("ENTER THE SMALLEST RADIUS AT THE PROMPT AND CONTINUE TO ENTER THE RADII") (a);
   put skip (1) edit ("IN ASCENDING ORDER.  AFTER THE LAST RADIUS HAS BEEN ENTERED, ENTER") (a);
   put skip (1) edit ("AN '0' AT THE NEXT PROMPT TO TERMINATE THE INPUT PROCEDURE.") (a);
   put skip (1) edit ("IF YOU ENTER A 0 FOR THE FIRST RADIUS THEN THE U.S.G.S.") (a);
   put skip (1) edit ("STANDARD RADII OF 0.25 MILE, 0.75 MILE, AND 3.0 MILES WILL BE USED.") (a);
   nxnum = 1;
radi:
   put skip (2) edit ("ENTER RADIUS FOR RELIABILITY CATEGORY",nxnum,".  ") (a);
   read file (usrin) into (keyin);
   if (keyin = "0" | keyin = "" | keyin = "0.") then go to opoint;
   put file (abfl) skip (1) edit (keyin) (a);
   nxnum = nxnum + 1;
   go to radi;
opoint:
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (2) edit ("NAME OF OBSERVED POINT FILE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "none" then go to skiprp;
eplta:
   put skip (2) edit ("ENTER A '1' IF YOU WANT ONLY THE Z VALUES PLOTTED,") (a);
   put skip (1) edit ("         '2' IF YOU WANT ONLY THE POINTIDS PLOTTED,") (a);
   put skip (1) edit ("         '3' IF YOU WANT BOTH PLOTTED.") (a);
   put skip (2) edit ("ENTER NUMBER: ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "1" & keyin ^= "2" & keyin ^= "3") then go to eplta;
   put file (abfl) skip (1) edit (keyin) (a);
```

```
skiprp:
   put skip (2) edit ("NAME OF GRIDDED INPUT FILE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
pmap:
   put skip (2) edit ("DO YOU WISH TO PLOT A RESOURCE MAP? ") (a);
   read file (usrin) into (keyin);
   keyin = substr (keyin,1,1);
   if (keyin ^= "y" & keyin ^= "n") then go to pmap;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "y" then go to mscale;
   go to boundf;
   if keyin = "n" then go to boundf;
mscale:
   put skip (2) edit ("ENTER MAP SCALE: ") (a);
   read file (usrin) into (calcv);
   put file (abfl) skip (1) edit (calcv) (a);
cvol:
   calcv = "";
   put skip (2) edit ("DO YOU WISH TO CALCULATE VOLUMES? (yes OR no): ") (a);
   read file (usrin) into (calcv);
   calcv = substr (calcv,1,1);
   if (calcv ^= "y" & calcv ^= "n") then go to cvol;
   put file (abfl) skip (1) edit (calcv) (a);
rmap:
   put skip (2) edit ("DO YOU WISH TO PLOT THE RESOURCE MAP WITH A CONTOUR SURFACE? (yes OR no): ")(a);
   read file (usrin) into (keyin);
   keyin = substr (keyin,1,1);
   if (keyin ^= "y" & keyin ^= "n") then go to rmap;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "n" then go to novol;
cint:
   put skip (2) edit ("CONTOUR INTERVAL: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to cint;
   put file (abfl) skip (1) edit (keyin) (a);
fbline:
   put skip (2) edit ("FREQUENCY OF BOLD LINES: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to fbline;
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (2) edit ("FOR CONTOUR SMOOTHING, ENTER A '1': ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
novol:
   put skip (2) edit ("TO SPECIFY GRAPHICS DEVICE, ENTER '0' FOR TEKTRONIX ONLY,") (a);
   put skip (1) edit ("                              ENTER '1' FOR TEKTRONIX AND CALCOMP,") (a);
   put skip (1) edit ("                          OR ENTER '2' FOR CALCOMP ONLY.") (a);
```

```
edvic:
   put skip (2) edit ("GRAPHICS DEVICE (0,1, or 2): ") (a);
   read file (usrin) into (gdvice);
   if (gdvice = "0" | gdvice = "1" | gdvice = "2") then go to gdevice;
   put skip (2) edit ("Enter 0, 1, or 2") (a);
   go to edvic;
gdevice:
   put file (abfl) skip (1) edit (gdvice) (a);
boundf:
   increc = 1;
boundf1:
   if increc = 1 then put skip (2) edit ("BOUNDARY FILE NAME:      ") (a);
   else put skip (2) edit ("BOUNDARY FILE NAME "; (substr(increc,8,3)), ": ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to boundf1;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "none" then go to enterx;
   increc=increc+1;
   if keyin ^= "none" then go to boundf1;
enterx:
   if gdvice = "0" then go to rtable;
   put skip (2) edit ("ENTER X AXIS TITLE (< 31 chars): ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
entery:
   put skip (2) edit ("ENTER Y AXIS TITLE (< 31 chars): ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
pfile:
   put skip (2) edit ("NAME OF PLOT FILE: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to pfile;
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (2) edit ("TO SPECIFY PLOTTING DEVICE, ENTER '1' FOR CALCOMP,") (a);
   put skip (1) edit ("                            ENTER '2' FOR VERSATEC") (a);
epdv:
   pdvice = "";
   put skip (2) edit ("ENTER PLOTTING DEVICE: ") (a);
   read file (usrin) into (pdvice);
   if (pdvice ^= "1" & pdvice ^= "2") then go to epdv;
   put file (abfl) skip (1) edit (pdvice) (a);
esym:
   put skip (2) edit ("ENTER SYMBOL SIZE FOR LINE AND POINT LABELS (DECIMAL INCHES): ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to esym;
   put file (abfl) skip (1) edit (keyin) (a);
   if gdvice = "1" then go to rtable;
   go to ecmmd;
```

```
rtable:
   if calcv = "n" then go to ecmmd;
   put skip (2) edit ("DO YOU WANT TO SAVE THE RESOURCE TABLE IN A FILE? ") (a);
   read file (usrin) into (keyin);
   keyin = substr (keyin,1,1);
   if (keyin ^= "y" & keyin ^= "n") then go to rtable;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "y" then go to yesrt;
   put skip (2) edit ("ENTER TITLE (LESS THAN 31 CH): ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
tonft:
   put skip (2) edit ("ENTER TONNAGE PER ACRE FOOT: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   put file (abfl) skip (1) edit ("") (a);
   if gdvice ^= "2" then put file (abfl) skip (1) edit ("") (a);
   go to ecmmd;
yesrt:
   put skip (2) edit ("WARNING- GARNET(reso) will not accept 'no' as an entry.") (a);
   put skip (2) edit ("ENTER RESOURCE TABLE FILE NAME: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
   put skip (2) edit ("ENTER TONNAGE PER ACRE FOOT: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
   put skip (2) edit ("ENTER STATE: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
   put skip (2) edit ("ENTER COUNTY: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
equad:
   put skip (2) edit ("ENTER QUAD: ") (a);
   read file (usrin) into (titlein);
   if titlein = "" then go to equad;
   put file (abfl) skip (1) edit (titlein) (a);
ebed:
   put skip (2) edit ("ENTER BED: ") (a);
   read file (usrin) into (titlein);
   if titlein = "" then go to ebed;
   put file (abfl) skip (1) edit (titlein) (a);
   put skip (2) edit ("ENTER TOWNSHIP: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
   put skip (2) edit ("ENTER N/S: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
   put skip (2) edit ("ENTER RANGE: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
```

```
   put skip (2) edit ("ENTER E/W: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
   put skip (2) edit ("ENTER THICKNESS CATEGORY: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
   put skip (2) edit ("ENTER OVERBURDEN CATEGORY: ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
   put file (abfl) skip (1) edit ("") (a);
   go to ecmmd;
contr:
   put file (abfl) skip (1) edit (cmmd) (a);
obfile:
   put skip (2) edit ("NAME OF OBSERVED POINT FILE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "none" then go to gif;
epltb:
   put skip (2) edit ("ENTER A '1' IF YOU WANT ONLY THE Z VALUES PLOTTED,") (a);
   put skip (1) edit ("        '2' IF YOU WANT ONLY THE POINTIDS PLOTTED,") (a);
   put skip (1) edit ("        '3' IF YOU WANT BOTH PLOTTED.") (a);
   put skip (2) edit ("ENTER NUMBER: ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "1" & keyin ^= "2" & keyin ^= "3") then go to epltb;
   put file (abfl) skip (1) edit (keyin) (a);
gif:
   put skip (2) edit ("NAME OF GRIDDED INPUT FILE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
cinter:
   put skip (2) edit ("CONTOUR INTERVAL: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to cinter;
   put file (abfl) skip (1) edit (keyin) (a);
freq:
   put skip (2) edit ("FREQUENCY OF BOLD LINES: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to freq;
   put file (abfl) skip (1) edit (keyin) (a);
csmooth:
   put skip (2) edit ("FOR CONTOUR SMOOTHING, ENTER A '1': ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "" & keyin ^= "1") then go to csmooth;
   put file (abfl) skip (1) edit (keyin) (a);
```

```
sgdevice:
   put skip (2) edit ("TO SPECIFY GRAPHICS DEVICE, ENTER '0' FOR TEKTRONIX ONLY,") (a);
   put skip (1) edit ("                          ENTER '1' FOR TEKTRONIX AND PLOTTER,") (a);
   put skip (1) edit ("                       OR ENTER '2' FOR PLOTTER ONLY.") (a);
edvic2:
   put skip (2) edit ("GRAPHICS DEVICE (0,1,2): ") (a);
   read file (usrin) into (gdvice);
   if (gdvice = "0" | gdvice = "1" | gdvice = "2") then go to gdvice2;
   put skip (2) edit ("Enter 0, 1, or 2") (a);
   go to edvic2;
gdvice2:
   put file (abfl) skip (1) edit (gdvice) (a);
   increc = 1;
boundf2:
   if increc = 1 then put skip (2) edit ("BOUNDARY FILE NAME:    ") (a);
   else put skip (2) edit ("BOUNDARY FILE NAME", (substr(increc,8,3)), ": ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to boundf2;
   put file (abfl) skip (1) edit (keyin) (a);
   if (keyin = "none" & gdvice = "0") then go to ecmmd;
   if keyin = "none" then go to entrx;
   increc = increc + 1;
   if increc ^= 99 then go to boundf2;
   if gdvice ^= "0" then go to entrx;
   put file (abfl) skip (1) edit ("") (a);
   go to ecmmd;
entrx:
   if gdvice = "0" then go to ecmmd;
   put skip (2) edit ("ENTER X AXIS TITLE (< 31 chars): ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
entry:
   put skip (2) edit ("ENTER Y AXIS TITLE (< 31 chars): ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
pltf:
   put skip (2) edit ("NAME OF PLOT FILE: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to pltf;
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (2) edit ("ENTER MAP SCALE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
```

```
specpd:
   put skip (2) edit ("TO SPECIFY PLOTTING DEVICE, ENTER '1' FOR CALCOMP,") (a);
   put skip (1) edit ("                              ENTER '2' FOR VERSATEC") (a);
   put skip (2) edit ("ENTER PLOTTING DEVICE: ") (a);
   read file (usrin) into (pdvice);
   if (pdvice ^= "1" & pdvice ^= "2") then go to specpd;
   put file (abfl) skip (1) edit (pdvice) (a);
esymb:
   put skip edit ("") (a);
   put skip edit ("ENTER SYMBOL SIZE FOR LINE AND POINT LABELS (DECIMAL INCHES): ")(a);
   read file (usrin) into (keyin);
   if keyin = "" then go to esymb;
   put file (abfl) skip (1) edit (keyin) (a);
   go to ecmmd;
windr:
   if bflag ^= 1 then go to nobwin;
nobwin:
   put file (abfl) skip (1) edit (cmmd) (a);
bframe:
   put skip (2) edit ("NAME OF GRIDDED INPUT FILE: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to bframe;
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (2) edit ("ENTER A '2' FOR KEYBOARD INPUT,") (a);
   put skip (1) edit ("        '3' FOR FILE INPUT.") (a);
einum:
   put skip (2) edit ("ENTER NUMBER: ") (a);
   read file (usrin) into (inum);
   if (inum ^= "2" & inum ^= "3") then go to einum;
   put file (abfl) skip (1) edit (inum) (a);
   go to ncurs;
cursori:
   keyin = "n";
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "n" then go to ncurs;
bfname:
   put skip (2) edit ("BOUNDARY FILE NAME:     ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to bfname;
   put file (abfl) skip (1) edit (keyin) (a);
opfile:
   put skip (2) edit ("NAME OF OBSERVED POINT FILE (< 9 chars): ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to opfile;
   tkey = keyin || "*";
   flag = search (substr (tkey,1), "*");
   if flag ^> 9 then go to good9;
   put skip (2) edit ("Must be <9 chars.") (a);
   if bflag = 1 then go to ebcmmd;
   go to opfile;
```

```
ncurs:
   if inum ^= "3" then go to wind2;
   put skip (1) edit ("ENTER WINDOW FILE NAME: ") (a
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (1) edit ("ENTER WINDOW NAME: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
wind2:
   if inum = "3" then go to bfchk;
   put skip (2) edit ("WEST: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to ncurs;
   put file (abfl) skip (1) edit (keyin) (a);
scurs:
   put skip (2) edit ("EAST: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to scurs;
   put file (abfl) skip (1) edit (keyin) (a);
ecurs:
   put skip (2) edit ("SOUTH: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to ecurs;
   put file (abfl) skip (1) edit (keyin) (a);
wcurs:
   put skip (2) edit ("NORTH: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to wcurs;
   put file (abfl) skip (1) edit (keyin) (a);
svwind:
   put skip (2) edit ("DO YOU WANT TO SAVE THIS WINDOW: ") (a);
   read file (usrin) into (keyin);
   keyin = substr (keyin,1,1);
   if (keyin ^= "n" & keyin ^= "y") then go to svwind;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "n" then go to bfchk;
eowfn:
   put skip (2) edit ("ENTER OUTPUT WINDOW FILE NAME: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to eowfn;
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (2) edit ("ENTER WINDOW NAME: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
bfchk:
   if bflag = 1 then go to ebcmmd;
   go to ecmmd;
good9:
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "none" then go to ecmmd;
   if wflag = 1 then go to ebcmmd;
```

```
epltc:
   put skip (2) edit ("ENTER A '1' IF YOU WANT ONLY THE Z VALUES PLOTTED,") (a);
   put skip (1) edit ("        '2' IF YOU WANT ONLY THE POINTIDS PLOTTED,") (a);
   put skip (1) edit ("        '3' IF YOU WANT BOTH PLOTTED.") (a);
   put skip (2) edit ("ENTER NUMBER: ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "1" & keyin ^= "2" & keyin ^= "3") then go to epltc;
   put file (abfl) skip (1) edit (keyin) (a);
   go to ecmmd;
locr:
   put file (abfl) skip (1) edit (cmmd) (a);
   put skip (2) edit ("NAME OF OBSERVED POINT FILE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "none" then go to locgin;
epltd:
   put skip (2) edit ("ENTER A '1' IF YOU WANT ONLY THE Z VALUES PLOTTED,") (a);
   put skip (1) edit ("        '2' IF YOU WANT ONLY THE POINTIDS PLOTTED,") (a);
   put skip (1) edit ("        '3' IF YOU WANT BOTH PLOTTED.") (a);
   put skip (2) edit ("ENTER NUMBER: ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "1" & keyin ^= "2" & keyin ^= "3") then go to epltd;
   put file (abfl) skip (1) edit (keyin) (a);
locgin:
   put skip (2) edit ("NAME OF GRIDDED INPUT FILE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
lfrac:
   put skip edit ("") (a);
   put skip edit ("ENTER FRACTION OF MAP TO BE PLOTTED OUTSIDE OF RECTANGULAR BOUNDARY: ")(a);
   read file (usrin) into (keyin);
   if keyin = "" then go to lfrac;
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (2) edit ("TO SPECIFY GRAPHICS DEVICE, ENTER '0' FOR TEKTRONIX ONLY,") (a);
   put skip (1) edit ("                            ENTER '1' FOR TEKTRONIX AND PLOTTER,") (a);
   put skip (1) edit ("                         OR ENTER '2' FOR PLOTTER ONLY.") (a);
ledvic2:
   put skip (2) edit ("GRAPHICS DEVICE (0,1,2): ") (a);
   read file (usrin) into (gdvice);
   if (gdvice = "0" | gdvice = "1" | gdvice = "2") then go to ldvic;
   put skip (2) edit ("Enter 0, 1, or 2") (a);
   go to ledvic2;
ldvic:
   put file (abfl) skip (1) edit (gdvice) (a);
   increc = 1;
lbnd2:
   if increc = 1 then put skip (2) edit ("BOUNDARY FILE NAME:    ") (a);
   else put skip (2) edit ("BOUNDARY FILE NAME", (substr(increc,8,3)), ": ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to lbnd2;
   put file (abfl) skip (1) edit (keyin) (a);
   if (keyin = "none" & gdvice = "0") then go to ecmmd;
   if keyin = "none" then go to lentrx;
   increc = increc + 1;
```

```
   if increc ^= 99 then go to lbnd2;
   if gdvice ^= "0" then go to lentrx;
   put file (abfl) skip (1) edit ("") (a);
   go to ecmmd;
lentrx:
   if gdvice = "0" then go to ecmmd;
   put skip (2) edit ("ENTER X AXIS TITLE (< 31 chars): ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
lentry:
   put skip (2) edit ("ENTER Y AXIS TITLE (< 31 chars): ") (a);
   read file (usrin) into (titlein);
   put file (abfl) skip (1) edit (titlein) (a);
lpltf:
   put skip (2) edit ("NAME OF PLOT FILE: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to lpltf;
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (2) edit ("ENTER MAP SCALE: ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
lspecpd:
   put skip (2) edit ("TO SPECIFY PLOTTING DEVICE, ENTER '1' FOR CALCOMP,") (a);
   put skip (1) edit ("                              ENTER '2' FOR VERSATEC") (a);
   put skip (2) edit ("ENTER PLOTTING DEVICE: ") (a);
   read file (usrin) into (pdvice);
   if (pdvice ^= "1" & pdvice ^= "2") then go to lspecpd;
   put file (abfl) skip (1) edit (pdvice) (a);
lesymb:
   put skip (2) edit ("ENTER SYMBOL SIZE FOR LINE AND POINT LABELS (DECIMAL INCHES): ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to lesymb;
   put file (abfl) skip (1) edit (keyin) (a);
   go to ecmmd;
qtr:
   put file (abfl) skip (1) edit (cmmd) (a);
   dot = search (substr (usrnm,1), ".");
   if dot = 0 then go to gdnm;
   tusrnm = "";
   tusrnm = (substr(usrnm,1,dot-1));
   usrnm = "";
   usrnm = tusrnm;
```

```
gdnm:
   reqi = usrnm || ".absin";
   reqo = usrnm || ".absout";
   dpareq = "dp " || reqi || " " || reqo;
   put skip (2) edit ("Like to add (",dpareq) (a);
   put skip edit ("(Enter  yes OR no): ") (a);
   read file (usrin) into (keyin);
   keyin = (substr (keyin,1,1));
   if keyin = "y" then put file (abfl) skip (1) edit (dpareq) (a);
   put file (abfl) skip (1) edit ("gr -set -string r -date -time -inc_vcpu -total_vcpu
 -inc_pf -total_pf -inc_mem_units -total_mem_units -inc_cost -total_cost") (a);
   put file (abfl) skip (1) edit ("logout") (a);
   usrnm = "";
   usrnm = reqi;
   put skip (3) edit (reqi) (a);
   put skip (2) edit ("Your absentee file has now been created") (a);
   put skip (1) edit ("with the suffix (.absin) added.") (a);
   put skip (3) edit ("(User should submit the request when ready.)") (a);
brk:
   put file (abfl) skip (1) edit ("") (a);
   close file (abfl);
   call rename ("abfl", usrnm);
endp:
   close file (usrin);
   put skip (2) edit ("") (a);
   end garnetab;
```