UNITED STATES DEPARTMENT OF INTERIOR

GEOLOGICAL SURVEY

PACERAB

A Computer Program to Create an Absentee File for Executing PACER

(Program to Analyze Coal Energy Resources).

By

William A. Scott

Open-File Report 82-217

1982

INTRODUCTION

PACERAB is an interactive program which was written to create a Multics absentee file to operate PACER, the PROGRAM to ANALYZE COAL ENERGY RESOURCES, data storage and retrieval system.

PURPOSE

By using PACERAB, the user can complete an entire PACER session, using considerably less time and expense. Instead of PACER being connected to a terminal and running interactively, PACERAB creates a file, containing PACER commands and input, that is connected to an absentee control segment which is run after normal working hours. PACERAB was designed for users experienced in using PACER, who can work through a session with little assistance.

USING PACERAB IN THE MULTICS ENVIRONMENT

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

```
lk >udd>URA>library>pacerab
```

If the user has Multics search rules established for operating PACER, the PACERAB program will be found automatically.

SESSION EXAMPLE

The sample run which follows illustrates the dialogue between user and the PACERAB (PACER absentee) program: User input is usually prompted by an 'Enter ...: ' or a numeric prompt and ended by pressing carriage return (C/R).

When entering data by a listed numeric or a letter prompt and when you wish to end the list, press only the C/R button.

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

Supplemental information from the author is in parentheses.

```
pacerab  (C/R)  (entered at Multics command level to begin program)

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

PACERAB CONTAINS OPTIONS TO SORT, DELETE, & PRINT MULTICS FILES.

ENTER NAME TO CALL YOUR ABSENTEE REQUEST-
   THE  PACERAB  PROGRAM  ADDS  ALL  SUFFIXES:  pabexample


ENTER 'pacer' AS IF YOU WERE EXECUTING THE PACER PROGRAM:  pacer


      WELCOME  TO THE  NCROS  PACER  ABSENTEE  PROGRAM.
      PROMPTS WILL BE VERY SIMILAR TO THOSE IN  'PACER'

PRESS CARRIAGE RETURN:  (c/r pressed)
```

```
LIKE TO SEE AVAILABLE DATA BASES? (ENTER  y OR n):  y

(A 'n' response will be followed by the 'ENTER DATABASE NAME:' prompt.)

AT THE CURRENT TIME, THE FOLLOWING DATA BASES ARE AVAILABLE:

USCOAL    - USGS COAL RESOURCES DATA

USALYT    - COAL ANALYTICAL DATA FOR SELECTED AREAS

USPEAT    - NOT AVAILABLE AT THIS TIME

USCHEM    - USGS GEOCHEMICAL ANALYTICAL DATA

USGEOL    - US GEOLOGICAL DATA

BMALYT    - USBM STANDARD US COAL ANALYTICAL DATA

USTRAT    - USGS STRATIGRAPHIC SEQUENCE DATA

BMRESBAS  - USBM SUMMARY RESERVE BASE AND AVG. ANAL.

USPET     - USGS PETROGRAPHIC DATA FOR SELECT. AREAS

ICHEM     - INT'L GEOCHEMICAL ANALYTICAL DATA

BEFORE ANY OF THESE  DATA BASES MAY BE  ACCESSED,
A DATA BASE FROM THE ABOVE LIST MUST BE SELECTED.

ENTER DATABASE NAME:  ustrat

(An unacceptable database name entry will respond with:
"'filename' is not an available database name', and repeat the prompt.)

LIKE TO SEE AVAILABLE COMMANDS? (ENTER  y OR n):  y

(A 'n' response will be followed by the 'ENTER COMMAND:  ' prompt.)
```

* THE COMMANDS WHICH MAY BE ISSUED (AND THEIR MEANING) ARE LISTED BELOW:

(DURING THIS SESSION, THE USER WILL OFTEN
BE ASKED TO ENTER ONE OF THESE COMMANDS.)

FILE - ALLOWS THE USER TO SELECT OR CHANGE THE DATA BASE TO BE USED.
COND - INITIATES THE REQUEST FOR RETRIEVAL CRITERIA TO BE ENTERED IN THE FORM: NAME REL VALUE
LOGI - INITIATES THE REQUEST FOR A LOGICAL EXPRESSION TO BE ENTERED USING LOGICAL OPERATORS.
SEAR - INITIATES THE SEARCH OF A FILE BASED UPON PREVIOUSLY ENTERED CONDITIONS AND LOGIC.
KEYS - INITIATES THE REQUEST FOR RETRIEVAL CRITERIA TO BE ENTERED IN THE FORM: BGINKEY,ENDKEY.
LIST - ALLOWS THE USER TO LIST SELECTED VALUES (VARIABLE NAMES WILL BE ASKED FOR) IN A FILE.
FUNC - PROVIDES FOR THE COMPUTATION OF FUNCTIONS ON ITEMS IN A DATA SET (OR FILE.)
DUMP - PRINTS ALL ITEMS PRESENT FOR EACH RECORD IN A SELECTED FILE. WAITS AFTER EACH N LINES.
STAT - ARRANGES SELECTED NUMERICAL DATA INTO STATPAC FORMAT.
DEFI - USED TO DEFINE NEW VARIABLE NAMES IN TERMS OF ORIGINAL ITEM NAMES (NAME=EXPRESSION)
TABL - USED TO COMPUTE U.S.G.S. STANDARD FORMAT TABLES. (TABLE ROUTINE HAS NOT BEEN FULLY UPDATED.)
QUIT - TERMINATES THE SYSTEM.

USER MAY ENTER 'help' AT THE 'ENTER COMMAND' PROMPT FOR A LIST OF COMMANDS.

ENTER COMMAND: file

(An unacceptable command entry will respond with:
"'command' not available', and repeat the prompt.)

ENTER DATABASE NAME: uschem

ENTER COMMAND: cond

A. bed eq pittsburgh

B. bed eq pittsburgh main

C. (c/r pressed)

ENTER COMMAND: logi

ENTER LOGIC: a+b

ENTER COMMAND: sear

ENTER INPUT FILE NAME: uschem

ENTER OUTPUT FILE NAME: searout

DO YOU WANT TO SEARCH ANOTHER INPUT FILE? (ENTER y OR n): n

(A 'y' response will return to the 'ENTER INPUT FILE NAME' prompt.)

```
ENTER COMMAND:  keys

          1.  1,2

          2.  3,4

          3.  (c/r pressed)


ENTER COMMAND:  list

ENTER NAME OF FILE:  searout

ENTER NUMBER OF LINES/PAGE (PRESS CR):  (c/r pressed)

3 TYPES OF LISTING ARE POSSIBLE:

C - COLUMN TYPE  (DEFAULT FORMAT)
F - COLUMN TYPE  (USER FORMAT)
R - ROW TYPE

SELECT  C, F, OR R:  c

WOULD YOU LIKE OUTPUT TO BE TO DISK? (ENTER  y OR n):  y

(A 'n' response will skip the next 2 disk file prompts.)

ENTER NAME OF DISK DATA SET TO BE CREATED:  listout

DO YOU WISH HEADINGS OUTPUTED TO THE DISK FILE?  y

(A 'n' response will also give the next prompt.)

ENTER THE LIST OF ITEM NAMES.

          1.  item1

          2.  item2

          3.  item3

          4.  (c/r pressed)
```

```
ENTER COMMAND:  func

ENTER NAME OF FILE:  searout

FUNCTIONS AVAILABLE AT THIS TIME ARE:
    MEAN     FIT     SUM

ENTER FUNCTION NAMES AND CORRESPONDING ARGUMENTS.

         1.  mean btu

         2.  fit bmash btu

         3.  sum estrank state colectr

         4.  (c/r pressed)

WOULD YOU LIKE OUTPUT TO BE TO DISK? (ENTER  y OR n):  y

(A 'n' response will skip the following disk prompt.)

ENTER NAME OF DISK DATA SET TO BE CREATED:  funcout

("FUNCOUT" disk  data set  will be created  before
           going to an unrelated set of commands.)

ENTER COMMAND:  dump

ENTER NAME OF FILE:  searout

ENTER NUMBER OF LINES/PAGE (PRESS C/R):

SHALL ALL CATEGORIES BE CONSIDERED? (ENTER  y OR n):  n

(A 'y' response will skip to the next 'ENTER COMMAND' prompt.)

EACH RECORD HAS BEEN DIVIDED INTO THE FOLLOWING GENERAL CATEGORIES:
       CAT. #  CAT. NAME
       ------  ---------
          1    LOCATION INFORMATION
          2    GEOLOGIC DESIGNATIONS
          3    BIBLIOGRAPHIC INFORMATION
          4    RANK INFORMATION
          5    ANALYSIS ID AND CODES
          o    PROXIMATE AND ULTIMATE DATA
          7    GEOCHEMICAL DATA (ASH,OXIDES)
          8    GEOCHEMICAL DATA (BA - IR)
          9    GEOCHEMICAL DATA (LA - RU)
         10    GEOCHEMICAL DATA (SB - ZR)
ENTER A LIST OF ASCENDING NUMBERS MATCHING YOUR CATEGORIES OF INTEREST

(ie. 1,3,5 OR 2-5):  1-3,5
```

```
ENTER COMMAND:  stat

ENTER LIST OF ITEM NAMES:

          1.  item1

          2.  item2

          3.  lastitem

          4.  (c/r pressed)

ENTER INPUT FILE NAME:  searout

ENTER OUTPUT STATPAC SEGMENT NAME
(WILL CONTAIN STATPAC DATASET(s)):  statout

ENTER NAME OF STATPAC DATASET:  statds

WHEN YOUR ABS REQUEST IS RUN,
THE STATPAC DATASET WILL BE CREATED WITH THE NAME:  statds

DO YOU HAVE ANOTHER DATABASE WITH THE SAME LIST OF ITEM NAMES
TO BE PLACED IN THIS SEGMENT. (ENTER yes OR no):  no

(A "y' response will return to the 'ENTER INPUT FILE NAME' prompt.)
```

```
ENTER COMMAND:  tabl


THE PACERAB-TABLE COMMAND HAS NOT BEEN FULLY UPDATED.

ENTER 'TABL' TO USE  *AS IT IS*
OR PRESS C/R FOR COMMAND PROMPT:  tabl

AT THE CURRENT TIME THE FOLLOWING LISTINGS ARE AVAILABLE:

1.  DSCRIP
2.  BMTAB
3.  OXIDTB
4.  CHTAB2
5.  STATWC

ENTER A LIST OF NUMBERS FOR DESIRED LISTINGS (ie. 1-3,5):  1-3,5

INPUT FILE NAME: searout

1 = WIDE CARRIAGE TERMINAL WHICH YOU ARE USING AT THIS TIME
2 = LINE PRINTER IN RESTON

WHERE DO YOU WANT THE TABLE PRINTED? (ENTER 2):  2

ENTER UNIQUE NAME FOR OUTPUT SEGMENT:  lpfile

HOW MANY PRINTS DO YOU WANT? (1-4):  1

ENTER NAME OF SAMPLE AREA (UP TO 32 CHARACTERS):  sample name

ENTER 1 OR 2 DIGIT NUMBER FO THE TABLE (WITH A DECIMAL):  1.


ENTER COMMAND:  defi


ENTER LIST OF NEW VARIABLE DEFINITIONS-

          1.  btuash=btu/bmash

          2.  save btuash

          3.  (c/r pressed)
```

```
(When user is ready to end session, enter 'quit').


ENTER COMMAND:  quit

(A listing of databases and files called follow)

INPUT FILES CALLED DURING THIS SESSION.
'DB-' INDICATES DATABASE:

DB- USTRAT
DB- USCHEM
      USCHEM
      SEAROUT


      (A listing of output files created follow)

MINIMUM LIST OF FILE(S) CREATED DURING THIS SESSION:
'DDS-' INDICATES (BINARY) DISK DATA SET.
'SIF-' INDICATES SEAR INPUT FILE.
'SDB-' INDICATES STAT DATABASE.

          SEAROUT
          LISTOUT
    DDS-  FUNCOUT
          STATOUT
          LPFILE

A PROMPT WILL APPEAR LATER TO DELETE UNWANTED FILES.

LIKE TO SORT ANY SAVED OUTPUT FILES? (ENTER  y OR n):  y

(A 'n' response will skip the 'SS command' prompts.)


A PROMPT TO ENTER EACH SS COMMAND WILL APPEAR.
ENTER REQUEST AS IF YOU WERE AT COMMAND LEVEL.
PRESS ONLY 'CR' TO END SS REQUEST.
(ie: ENTER SS COMMAND:  ss file -asc -fl 20 8 30 10)
THESE LINES  WILL BE  PLACED IN  THE ABSENTEE  FILE.

ENTER SS COMMAND:  ss listout -fl 1 10


ENTER SS COMMAND:  (c/r pressed)
```

```
LIKE TO DL AND/OR DP ANY FILES USED DURING THIS SESSION ?
(ENTER  y OR n):  y

(A "n' response will skip the following "ENTER REQUEST' prompts.)


ENTER MULTICS COMMANDS TO D_PRINT AND/OR DELETE FILES.
PRESS ONLY "CR' TO END REQUESTS.

ENTER REQUEST:  dp -dl listout funcout


ENTER REQUEST:  (c/r pressed)


pabexample.absin

LIKE TO ADD "dp pabexample.absin pabexample.absout" TO REQUEST?
(ENTER  y OR n):  y

(A "n' response will not add this Multics command to the absentee file.)


YOUR ABSENTEE FILE HAS NOW BEEN CREATED
WITH THE SUFFIX (.ABSIN) ADDED.
```

ENTERING THE ABSENTEE REQUEST

(USER SHOULD SUBMIT THE REQUEST WHEN READY.)

example:

```
ear pabexample.absin -queue 4
```

CREATED ABSENTEE FILE

the following (pabexample.absin) file was created during this session:

```
pacer
23
ustrat
file
uschem
cond
bed eq pittsburgh
bed eq pittsburgh main

logi
a+b
sear
uschem
searout
n
keys
1,2
3,4

list
searout

c
y
listout
y
item1
item2
item3

func
searout
mean btu
fit bmash btu
sum estrank state colectr

y
funcout
dump
searout

n
1-3,5
stat
item1
item2
lastitem

searout
statout
statds
no
tabl
tabl
```

```
1-3,5
searout
2
lpfile
1
sample name
1.
defi
btuash=btu/bmash
save btuash

quit
ss listout -fl 1 10
dp -dl listout funcout
dp pabexample.absin pabexample.absout
logout
```

```
        (stringrange):
        (subscriptrange):
pacerab:
        proc;
    dcl (abfl)    file;
    dcl (outfil)  file;
    dcl (infil)   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;
    open file (outfil)  stream output;
    open file (infil)   stream output;
    dcl delete entry options (variable);
    dcl new_proc entry;
    dcl dp entry options (variable);
    dcl quit cond;
    dcl endfile cond;
    dcl error cond;
    dcl lstop  char (12) varying init ("");
    dcl stin   char (12) varying init ("");
    dcl dbname char (24) varying init ("");
    dcl dds    char (24) varying init ("");
    dcl sif    char (24) varying init ("");
    dcl datab  char (24) varying init ("");
    dcl usrnm  char (16) varying init ("");
    dcl cmmd   char (16) varying init ("");
    dcl cmmd4  char (6)  varying init ("");
    dcl tlog   char (16) varying init ("");
    dcl tcmmd  char (16) varying init ("");
    dcl achar  char (4)  varying init ("");
    dcl newc   char (16) varying init ("");
    dcl keyin  char (64) varying init ("");
    dcl chkf   char (16) varying init ("");
```

```
    dcl svchkf char (16) varying init ("");
    dcl techk  char (48) varying init ("");
    dcl qtin   char (16) varying init ("");
    dcl reqi   char (48) varying init ("");
    dcl reqo   char (48) varying init ("");
    dcl tusrnm char (48) varying init ("");
    dcl lsname char (48) varying init ("");
    dcl dpareq char (48) varying init ("");
    dcl ssop   char (8)  varying init ("");
    dcl ssln   char (36) varying init ("");
    dcl dpop   char (8)  varying init ("");
    dcl tdpnums char (24) varying init ("");
    dcl dpnums char (24)  varying init ("");
    dcl dpline char (140) varying init ("");
    dcl dpreq  char (140) varying init ("");
    dcl searo  char (16)  varying init ("");
    dcl atabl  char (26);
    atabl = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
    on error go to nofl;
    on endfile (outfil) go to lastnm;
    on quit go to brk;
    dcl (nxnum,freq,filrd,dp2,dot,lcnt,echk,scflag,cflag,nxc,filc,totf,dpc,iflag) fixed bin;
    nxnum,freq,filrd,dp2,dot,lcnt,echk,scflag,nxc,filc,totf,dpc,iflag,cflag,nxc,filc,totf,
    dpc,iflag = 0;
    put skip (1) edit ("'pacerab'  CREATES A MULTICS ABSENTEE INPUT FILE CONTAINING") (a);
    put skip (1) edit ("PACER COMMANDS AND FUNCTIONS THAT WOULD BE ENTERED IF PACER") (a);
    put skip (1) edit ("WERE  EXECUTED  INTERACTIVELY.   ") (a);
    put skip (1) edit ("PACERAB  contains options to  ss, dl, & dp  MULTICS  files.") (a);
ent2:
    put skip (2) edit ("ENTER NAME TO CALL YOUR ABSENTEE REQUEST-") (a);
    put skip (1) edit ("   The  pacerab  program  adds  all  suffixes:  ") (a);
    read file (usrin) into (usrnm);
    if usrnm = "" then go to ent2;
```

```
begin:
   put skip (2) edit ("Enter 'pacer' as if you were executing the pacer program:  ") (a);
   read file (usrin) into (stin);
   if stin ^= "pacer" then go to begin;
   put file (abfl) edit (stin) (a);
   put skip (2) edit ("          Welcome  to the  NCRDS  PACER  ABSENTEE  program.") (a);
   put skip (1) edit ("          Prompts will be very similar to those in  'pacer'") (a);
   put skip (2) edit ("          Press carriage return:  ") (a);
   read file (usrin) into (lstop);
   put file (abfl) skip (1) edit (lstop) (a);
dblis:
   put skip (2) edit ("Like to see available data bases? (Enter  y OR n):  ") (a);
   read file (usrin) into (lstop);
   if (lstop = "n" | lstop = "no") then go to skdb;
   put skip (2) edit ("At the current time, the following data bases are available:") (a);
   put skip (2) edit ("uscoal    - USGS coal resources data") (a);
   put skip (2) edit ("usalyt    - coal analytical data for selected areas") (a);
   put skip (2) edit ("uspeat    - not available at this time") (a);
   put skip (2) edit ("uschem    - USGS geochemical analytical data") (a);
   put skip (2) edit ("usgeol    - US geological data") (a);
   put skip (2) edit ("bmalyt    - USBM standard us coal analytical data") (a);
   put skip (2) edit ("ustrat    - USGS stratigraphic sequence data") (a);
   put skip (2) edit ("bmresbas - USBM Summary Reserve Base and Avg. Anal.") (a);
   put skip (2) edit ("uspet     - USGS petrographic data for select. areas") (a);
   put skip (2) edit ("ichem     - int'l geochemical analytical data") (a);
   put skip (3) edit ("Before any of these data bases may be accessed,") (a);
   put skip (1) edit ("a data base from the above list must be selected.") (a);
skdb:
   datab, dbname, dds = "";
   put skip (3) edit ("Enter database name:  ") (a);
   read file (usrin) into (dbname);
   if dbname = "" then go to skdb;
   if (dbname = "uscoal" | dbname = "usalyt" | dbname = "uspeat")  then go to gddb;
   if (dbname = "uschem" | dbname = "usgeol" | dbname = "bmalyt")  then go to gddb;
   if (dbname = "ustrat" | dbname = "bmresbas" | dbname = "uspet") then go to gddb;
   put skip (2) edit (dbname," is not an available data base name.") (a);
   go to dolis;
```

```
gddb:
 put file (abfl) skip (1) edit (dbname) (a);
 datab = "DB- " || dbname;
 put file (infil) skip (1) edit (datab) (a);
 cmmd = "";
 put skip (2) edit ("Like to see available commands? (Enter  y OR n):  ") (a);
 read file (usrin) into (cmmd);
 if (cmmd ^= "y" & cmmd ^= "yes") then go to skcmd;
seec:
 put skip (2) edit ("* The commands which may be issued (and their meaning) are listed below:")(a);
 put skip (2) edit ("file - Allows the user to select or change the data base to be used.") (a);
 put skip (1) edit ("cond - Initiates the request for retrieval criteria to be entered in") (a);
 put skip (1) edit ("       the form: name rel value") (a);
 put skip (1) edit ("logi - Initiates the request  for a logical expression to be entered") (a);
 put skip (1) edit ("       using logical operators.") (a);
 put skip (1) edit ("sear - Initiates the search  of a file based upon previously entered") (a);
 put skip (1) edit ("       conditions and logic.") (a);
 put skip (1) edit ("keys - Initiates the request for retrieval criteria to be entered in") (a);
 put skip (1) edit ("       the form: bginkey,endkey.") (a);
 put skip (1) edit ("list - Allows the user to list selected values  (variable names will") (a);
 put skip (1) edit ("       be asked for) in a file.") (a);
 put skip (1) edit ("func - Provides for the computation  of functions on items in a data") (a);
 put skip (1) edit ("       set (or file.)") (a);
 put skip (1) edit ("dump - Prints all items present  for each record in a selected file.") (a);
 put skip (1) edit ("       waits after each n lines.") (a);
 put skip (1) edit ("stat - Arranges selected numerical data into STATPAC format.") (a);
 put skip (1) edit ("tabl - Used to compute U.S.G.S. standard format tables.") (a);
 put skip (1) edit ("       (table routine has not been fully updated.)") (a);
 put skip (1) edit ("defi - Used to define new variable names  in terms of original item") (a);
 put skip (1) edit ("       names (name=expression)") (a);
 put skip (1) edit ("quit - Terminates the system.") (a);
 put skip (2) edit ("User may enter 'help' at the 'ENTER COMMAND' prompt a list of commands.")(a);
skcmd:
 tlog = "";
 filc, totf = 0;
```

```
ecmmd:
   cmmd = "";
   put skip (3) edit ("Enter command:  ") (a);
   read file (usrin) into (cmmd);
   if cmmd = "help" then go to seec;
   if cmmd = "" then go to ecmmd;
   tcmmd = cmmd ||  " xxxx";
   cmmd4 = substr (tcmmd,1,4);
   cmmd = cmmd4;
   if cmmd = "logi" then tlog = "logi";
   if cmmd = "keys" then tlog = "keys";
   if cmmd ^= "file" then go to con;
   go to filr;
con:
   if cmmd ^= "cond" then go to log;
   go to condr;
log:
   if cmmd ^= "logi" then go to srch;
   go to logr;
srch:
   if cmmd ^= "sear" then go to kes;
   go to srchr;
kes:
   if cmmd ^= "keys" then go to lis;
   go to keyr;
lis:
   if cmmd ^= "list" then go to fun;
   go to listr;
fun:
   if cmmd ^= "func" then go to dum;
   go to funr;
dum:
   if cmmd ^= "dump" then go to sta;
   go to dumr;
sta:
   if cmmd ^= "stat" then go to tab;
   go to statr;
```

```
tab:
   if cmmd ^= "tabl" then go to def;
   go to tabr;
def:
   if cmmd ^= "defi" then go to qt;
   go to defr;
qt:
   if cmmd = "quit" then go to qtr;
   put skip (2) edit (cmmd, " command not available") (a);
   go to ecmmd;
filr:
   put file (abfl) skip (1) edit (cmmd) (a);
edb:
   datab, dbname, dds = "";
   put skip (2) edit ("Enter database name:  ") (a);
   read file (usrin) into (dbname);
   if dbname = "" then go to edb;
   put file (abfl) skip (1) edit (dbname) (a);
   datab = "DB- " || dbname;
   put file (infil) skip (1) edit (datab) (a);
   go to ecmmd;
condr:
   put file (abfl) skip (1) edit (cmmd) (a);
   nxc = 1;
   achar = "";
nxcnd:
   achar = substr (atabl,nxc,1);
   newc = achar || ".  ";
   put skip (1) edit (newc) (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   if (keyin = "" | nxc = 26) then go to ecmmd;
   nxc = nxc + 1;
   go to nxcnd;
```

```
logr:
   put file (abfl) skip (1) edit (cmmd) (a);
logi:
   put skip (1) edit ("Enter logic:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to logi;
   put file (abfl) skip (1) edit (keyin) (a);
   go to ecmmd;
srchr:
   sif,keyin,searo = "";
   if (tlog = "logi" | tlog = "keys") then go to puts;
   put skip (1) edit ("Logic must be supplied before a retrieval can be made.") (a);
   go to ecmmd;
puts:
   put file (abfl) skip (1) edit (cmmd) (a);
nxsr:
   put skip (1) edit ("Enter input file name:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to nxsr;
   put file (abfl) skip (1) edit (keyin) (a);
   put file (infil) skip (1) edit (keyin) (col(7),a);
   sif = "SIF- " || keyin;
   if filc = 0 then put file (outfil) edit (sif) (a);
   else put file (outfil) skip (1) edit (sif) (a);
   totf = totf + 1;
   filc = 1;
entf:
   if searo ^= "" then go to nosout;
   put skip (1) edit ("Enter output file name:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to entf;
   searo = keyin;
   put file (abfl) skip (1) edit (keyin) (a);
   if filc = 0 then put file (outfil) edit (keyin) (col(6),a);
   else put file (outfil) skip (1) edit (keyin) (col(6),a);
   totf = totf + 1;
   filc = 1;
nosout:
   if dbname ^= "ustrat" then go to skbed;
```

```
topop:
   put skip (1) edit ("Do you want only the top unit of the bed? (y OR n):  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to topop;
   put file (abfl) skip (1) edit (keyin) (a);
skbed:
   put skip (1) edit ("Do you want to search another input file? (Enter  y OR n):  ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "y" & keyin ^= "yes") then put file (abfl) skip (1) edit ("n") (a);
   else put file (abfl) skip (1) edit (keyin) (a);
   if (keyin = "y" | keyin = "yes") then go to nxsr;
   else go to ecmmd;
keyr:
   put file (abfl) skip (1) edit (cmmd) (a);
   nxc = 1;
nxki:
   cflag,scflag = 0;
   newc = nxc || ".  ";
   put skip (1) edit (newc) (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to ekeyin;
   cflag = search (substr (keyin,1), ",");
   scflag = search (substr (keyin,1), ";");
   if (cflag ^= 0 & scflag ^= 0) then go to ekeyin;
   put list ("Keys must be in pairs, separated with a ','") skip (1);
   put list ("or need a ';' for a 1 field entry.") skip (1);
   go to nxki;
ekeyin:
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "" then go to ecmmd;
   nxc = nxc + 1;
   go to nxki;
```

```
listr:
   put file (abfl) skip (1) edit (cmmd) (a);
lisi:
   put skip (1) edit ("Enter name of file:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to lisi;
   put file (abfl) skip (1) edit (keyin) (a);
   put file (infil) skip (1) edit (keyin) (col(7),a);
   put skip (1) edit ("Enter number of lines/page (press CR):  ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit ("") (a);
   put skip (2) edit ("3 Types of listing are possible:") (a);
   put skip (2) edit ("c - Column type  (default format)") (a);
   put skip (1) edit ("f - Column type  (user format) ") (a);
   put skip (1) edit ("r - Row type") (a);
ltype:
   put skip (2) edit ("Select  c, f, or r:  ") (a);
   read file (usrin) into (keyin);
   if (keyin = "c" | keyin = "f" | keyin = "r") then go to goodl;
   put skip (1) edit ("Your reply was not understood.") (a);
   go to ltype;
goodl:
   put file (abfl) skip (1) edit (keyin) (a);
   put skip (1) edit ("Would you like output to be to disk? (Enter  y OR n):  ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "y" & keyin ^= "yes") then put file (abfl) skip (1) edit ("n") (a);
   else put file (abfl) skip (1) edit (keyin) (a);
   if (keyin ^= "y" & keyin ^= "yes") then go to skls;
entrdd:
   keyin = "";
   put skip (2) edit ("Enter name of disk data set to be created:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to entrdd;
   put file (abfl) skip (1) edit (keyin) (a);
   dds = "DDS- " || keyin;
   if filc = 0 then put file (outfil) edit (dds) (a);
   else put file (outfil) skip (1) edit (dds) (a);
   totf = totf + 1;
   filc = 1;
```

```
headi:
put skip (2) edit ("Do you wish headings outputed to the disk file?  ") (a);
read file (usrin) into (keyin);
if keyin = "" then go to headi;
put file (abfl) skip (1) edit (keyin) (a);
if (keyin ^= "y" & keyin ^= "yes") then go to skls;
if keyin ^= "JUMP" then go to skls;
put skip (2) edit ("Would you like to see a list of the item names? (Enter  y OR n):  ") (a);
read file (usrin) into (keyin);
if (keyin ^= "y" & keyin ^= "yes") then put file (abfl) skip (1) edit ("n") (a);
else put file (abfl) skip (1) edit (keyin) (a);
if (keyin ^= "y" & keyin ^= "yes") then go to skls;
put skip(2) edit ("The original item names are:") (a);
put skip (1) ("") (a);
put skip edit ("state   county  latitud ns      longtud ew      aapgprv coalprv region  field")(a);
put skip edit ("distrct formatn group   bed     member  zone    depth   sampthk system  series")(a);
put skip edit ("locname quad    colectr fieldid date    estrank majrank minrank anida   anidn")(a);
put skip edit ("labcode samptyp analtyp valrep  othert  btu     ashdef  ashsof  ashfld  freswel")(a);
put skip edit ("moistur volmat  fixedc  bmash   hydrogn carbon  nitrogn oxygen  sulfur  sulfate")(a);
put skip edit ("sulfpyr sulfory adloss  usgsash sio2    al2o3   cao     mgo     na2o    k2o") (a);
put skip edit ("fe2o3   mno     tio2    p2o5    so3     cl      ag      as      au      b") (a);
put skip edit ("ba      be      bi      br      cd      ce      co      cr      cs      cu") (a);
put skip edit ("dy      er      eu      f       ga      gd      ge      hf      hg      ho") (a);
put skip edit ("in      ir      la      li      lu      mn      mo      nb      nd      ni") (a);
put skip edit ("os      pb      pd      pr      pt      rb      re      rh      ru      sb") (a);
put skip edit ("sc      se      sm      sn      sr      ta      tb      te      th      tl") (a);
put skip edit ("tm      u       v       w       y       yb      zn      zr      secure  lastupd")(a);
put skip edit ("") (a);
skls:
   nxc = 1;
   if lcnt = 0 then put skip (2) edit ("Enter the list of item names.") (a);
   else go to nwlis;
   lcnt = lcnt + 1;
nxitm:
   newc = nxc || ".  ";
   put skip (2) edit (newc) (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "" then go to ecmmd;
   nxc = nxc + 1;
   go to nxitm;
```

```
nwlis:
   nxc= 1;
   keyin = "";
   put skip (2) edit ("Do you wish to enter a new list of names ?  ") (a);
   read file (usrin) into (keyin);
   if (keyin = "no" | keyin = "n") then put file (abfl) skip (1) edit ("n") (a);
   else put file (abfl) skip (1) edit ("y") (a);
   if (keyin ^= "n" & keyin ^= "no") then go to nxitm;
   else go to ecmmd;
funr:
   put file (abfl) skip (1) edit (cmmd) (a);
efin:
   chkf, techk, newc, svcnkf = "";
   put skip (1) edit ("Enter name of file:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to efin;
   put file (abfl) skip (1) edit (keyin) (a);
   put file (infil) skip (1) edit (keyin) (col(7),a);
   nxc = 1;
efun:
   put skip (1) edit ("Functions available at this time are:") (a);
   put skip (1) edit ("    mean    fit    sum") (a);
   put skip (2) edit ("Enter function names and corresponding arguments.") (a);
   put skip (2) edit ("") (a);
nxs:
   newc = nxc || ".  ";
   put skip (1) edit (newc) (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to endf;
   techk = keyin || "ZZZZ";
   echk = search (substr (techk,1), "ZZ");
   if echk < 5 then go to nxs;
   chkf = substr(keyin,1,4);
   if svchkf = chkf then go to twice;
   if (chkf = "mean" | chkf = "fit " | chkf = "sum ") then go to gdchk;
   else go to efun;
twice:
   put skip(2) edit (chkf, " may appear only once during each execution of the function command")(a);
   go to nxs;
```

```
gdchk:
   put file (abfl) skip (1) edit (keyin) (a);
   svchkf = chkf;
   nxc = nxc + 1;
   go to nxs;
endf:
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "" then go to fdk;
   if (chkf = "mean" | chkf = "fit ") then go to sksum;
fdk:
   put skip (2) edit ("Would you like output to be to disk? (Enter  y OR n):  ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "y" & keyin ^= "n") then go to fdk;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "n" then go to ecmmd;
fds:
   put skip (1) edit ("Enter name of disk data set to be created:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to fds;
   put file (abfl) skip (1) edit (keyin) (a);
   dds = "DDS- " || keyin;
   if filc = 0 then put file (outfil) edit (dds) (a);
   else put file (outfil) skip (1) edit (dds) (a);
sksum:
   totf = totf + 1;
   filc = 1;
   go to ecmmd;
```

```
dumr:
   put file (abfl) skip (1) edit (cmmd) (a);
edumi:
   put skip (1) edit ("Enter name of file:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to edumi;
   put file (abfl) skip (1) edit (keyin) (a);
   put file (infil) skip (1) edit (keyin) (col(7),a);
   put skip (1) edit ("Enter number of lines/page (press CR):  ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit ("") (a);
   put skip (1) edit ("Shall all categories be considered? (Enter  y OR n):  ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "n" & keyin ^= "no") then put file (abfl) skip (1) edit ("y") (a);
   else put file (abfl) skip (1) edit (keyin) (a);
   if (keyin = "y" | keyin = "yes") then go to ecmmd;
   put skip (1) edit ("Each record has been divided into the following general categories:") (a);
   put skip (1) edit ("       Cat. #  Cat. name") (a);
   put skip (1) edit ("       ------  ---------") (a);
   put skip (2) edit ("          1    location information") (a);
   put skip (1) edit ("          2    geologic designations") (a);
   put skip (1) edit ("          3    bibliographic information") (a);
   put skip (1) edit ("          4    rank information") (a);
   put skip (1) edit ("          5    analysis id and codes") (a);
   put skip (1) edit ("          6    proximate and ultimate data") (a);
   put skip (1) edit ("          7    geochemical data (ash,oxides)") (a);
   put skip (1) edit ("          8    geochemical data (ba - ir)") (a);
   put skip (1) edit ("          9    geochemical data (la - ru)") (a);
   put skip (1) edit ("         10    geochemical data (sb - zr)") (a);
   put skip (2) edit ("Enter a list of ascending numbers matching your categories of interest.") (a);
   put skip (2) edit ("(i.e. 1,3,5 or 2-5):  ") (a);
numin:
   put skip (1) edit ("Enter:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to numin;
   put file (abfl) skip (1) edit (keyin) (a);
   go to ecmmd;
```

```
statr:
   dds = "";
   nxc = 1;
   put skip (2) edit ("Enter list of item names (or a '?' for more information.)  ") (a);
nxiname:
   newc = nxc || ".  ";
   put skip (2) edit (newc) (a);
   read file (usrin) into (keyin);
   if keyin = "?" then go to help;
   if nxc = 1 then put file (abfl) skip (1) edit (cmmd) (a);
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "" then go to statin;
   nxc = nxc + 1;
   go to nxiname;
statin:
   put skip (2) edit ("Enter input file name:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to statin;
   put file (abfl) skip (1) edit (keyin) (a);
   put file (infil) skip (1) edit (keyin) (col (7),a);
statout:
   put skip (2) edit ("Enter output Statpac segment name (will contain Statpac dataset(s)):  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to statout;
   dds = "SDB- " || keyin;
   if filc = 0 then put file (outfil) edit (dds) (a);
   else put file (outfil) skip (1) edit (dds) (a);
   totf = totf + 1;
   filc = 1;
   put file (abfl) skip (1) edit (keyin) (a);
statds:
   put skip (2) edit ("Enter name of Statpac dataset:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to statds;
   put file (abfl) skip (1) edit (keyin) (a);
   put file (infil) skip (1) edit (keyin) (col (7),a);
   put skip (2) edit ("When your abs request is run,") (a);
   put skip (1) edit ("The Statpac dataset will be created with the name:  ", keyin) (a);
```

```
nxtds:
   put skip(2) edit ("Do you have another dataset with the same list of item names to be placed")(a);
   put skip(1) edit ("in this segment. (Enter yes OR no):  ") (a);
   read file (usrin) into (keyin);
   if (keyin ^= "yes" & keyin ^= "no") then go to nxtds;
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "yes" then go to statin;
   go to ecmmd;
help:
   put skip (2) edit ("No help is available at this time.") (a);
   go to ecmmd;
tabr:
   put skip (2) edit ("The pacerab-table command has not been fully updated.") (a);
   put skip (1) edit ("Please let me know if it will be helpful to have.") (a);
   put skip (1) edit ("Bill Scott x 6451") (a);
   put skip (1) edit ("") (a);
   put skip (1) edit ("Enter 'tabl' to use  *as it is*") (a);
   put skip (1) edit ("OR PRESS C/R for command prompt:  ") (a);
   read file (usrin) into (cmmd);
   if cmmd ^= "tabl" then go to ecmmd;
   put file (abfl) skip (1) edit (cmmd) (a);
   put skip (1) edit ("At the current time the following listings are available:  ") (a);
   put skip (2) edit ("1.  dscrip") (a);
   put skip (1) edit ("2.  bmtab")  (a);
   put skip (1) edit ("3.  oxidtb") (a);
   put skip (1) edit ("4.  chtabz") (a);
   put skip (1) edit ("5.  statwc") (a);
tnumi:
   put skip (2) edit ("Enter a list of numbers for desired listings (ie. 1-3,5):  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to tnumi;
   put file (abfl) skip (1) edit (keyin) (a);
```

```
tfili:
   put skip (1) edit ("Input file name: ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to tfili;
   put file (abfl) skip (1) edit (keyin) (a);
   put file (infil) skip (1) edit (keyin) (col(7),a);
   put skip (1) edit ("1 = wide carriage terminal which you are using at this time") (a);
   put skip (1) edit ("2 = line printer in Reston") (a);
   put skip (2) edit ("Where do you want the table printed? (Enter 2):  ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit ("2") (a);
entu:
   put skip (1) edit ("Enter unique name for output segment:  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to entu;
   if filc = 0 then put file (outfil) edit (keyin) (col(6),a);
   else put file (outfil) skip (1) edit (keyin) (col(6),a);
   totf = totf + 1;
   filc = 1;
   put file (abfl) skip (1) edit (keyin) (a);
entp:
   put skip (1) edit ("How many prints do you want? (1-4):  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to entp;
   dpc = keyin;
   if dpc > 4 then dpc = 4;
   put file (abfl) skip (1) edit (dpc) (a);
sami:
   put skip (1) edit ("Enter name of sample area (up to 32 characters):  ") (a);
   read file (usrin) into (keyin);
   if keyin = "" then go to sami;
   put file (abfl) skip (1) edit (keyin) (a);
   dot = 0;
dig:
   put skip (1) edit ("Enter 1 or 2 digit number fo the table (with a decimal):  ") (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   dot = search (substr(keyin,1), ".");
   if dot = 0 then go to dig;
   go to ecmmd;
```

```
defr:
   put file (abfl) skip (1) edit (cmmd) (a);
   nxc = 1;
   put skip (2) edit ("Enter list of new variable definitions-") (a);
   put skip (1) edit ("") (a);
nxdf:
   newc = nxc || ".  ";
   put skip (1) edit (newc) (a);
   read file (usrin) into (keyin);
   put file (abfl) skip (1) edit (keyin) (a);
   if keyin = "" then go to ecmmd;
   nxc = nxc + 1;
   go to nxdf;
qtr:
   put file (abfl) skip (1) edit (cmmd) (a);
   put file (infil) skip (1) edit ("END") (a);
   put file (infil) skip (1) edit ("") (a);
   if filc = 0 then put file (outfil) edit ("END") (a);
   else put file (outfil) skip (1) edit ("END") (a);
   put file (outfil) skip (1) edit ("") (a);
   close file (infil);
   close file (outfil);
   open file (infil) environment (stringvalue) record input
   title ("record_stream_ -target vfile_ infil");
   put skip (1) edit ("Input files called during this session.") (a);
   put skip (1) edit ("'DB-' indicates database:") (a);
   open file (outfil) environment (stringvalue) record input
   title ("record_stream_ -target vfile_ outfil");
   lsname = "";
rdinp:
   read file (infil) into (lsname);
   if lsname = "END" then go to outlst;
   put skip (1) edit (lsname) (a);
   go to rdinp;
outlst:
   nxc = 1;
   if totf > 0 then go to hvfl;
   else go to nofl;
```

```
hvfl:
   put skip (3) edit ("Minimum list of file(s) created during this session:") (a);
   put skip (1) edit ("'DDS-' indicates (binary) disk data set.") (a);
   put skip (1) edit ("'SIF-' indicates SEAR input file.") (a);
   put skip (1) edit ("'SDB-' indicates STAT dataset.") (a);
   put skip (1) edit ("") (a);
rdlst:
   read file (outfil) into (lsname);
   if lsname = "END" then go to lastnm;
   newc = nxc || ".  ";
   put skip (1) edit ("    ",lsname) (a);
   nxc = nxc + 1;
   go to rdlst;
lastnm:
   qtin = "";
   nxnum,freq,filrd,dot,dp2,iflag = 0;
   put skip (2) edit ("A prompt will appear later to delete unwanted files.") (a);
ssopt:
   put skip (2) edit ("Like to SORT any saved output files? (Enter  y OR n):  ") (a);
   read file (usrin) into (ssop);
   if ssop = "" then go to ssopt;
   else if (ssop ^= "y" & ssop ^= "yes") then go to dlop;
   put skip (2) edit ("A prompt to enter each SS command will appear.") (a);
   put skip (1) edit ("Enter request as if you were at command level.") (a);
   put skip (1) edit ("Press only 'CR' to end SS request.") (a);
   put skip (1) edit ("(ie: Enter SS command:  ss file -asc -fl 20 8 30 10)") (a);
   put skip (1) edit ("These lines  will be  placed in  the absentee  file.") (a);
entss:
   put skip (2) edit ("Enter SS command:  ") (a);
   read file (usrin) into (ssln);
   if ssln = "" then go to dlop;
   put file (abfl) skip (1) edit (ssln) (a);
   go to entss;
dlop:
   put skip (2) edit ("Like to dl and/or dp any files used during this session ? ") (a);
   put skip (1) edit ("(Enter  y OR n):  ") (a);
   read file (usrin) into (dpop);
   if dpop = "" then go to dlop;
   else if (dpop ^= "y" & dpop ^= "yes") then go to nofl;
   put skip (2) edit("Enter MULTICS commands to d_print and/or delete files.") (a);
   put skip (1) edit ("Press only 'CR' to end requests.") (a);
```

```
ereqs:
   put skip (2) edit ("Enter Request:  ") (a);
   read file (usrin) into (dpline);
   if dpline = "" then go to nofl;
   put file (abfl) skip (1) edit (dpline) (a);
   go to ereqs;
nofl:
   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 edit (") to request ? (Enter  y OR n):  ") (a);
   read file (usrin) into (dpop);
   if (dpop = "y" | dpop = "yes") then put file (abfl) skip (1) edit (dpareq) (a);
   put file (abfl) skip (1) edit ("logout") (a);
   put file (abfl) skip (1) edit ("") (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:
   close file  (outfil);
   close file  (abfl);
   call rename ("abfl", usrnm);
   call delete ("outfil");
   call delete ("infil");
endp:
   close file (usrin);
   put skip  (2) edit  ("")  (a);
   call new_proc;
   end pacerab;
```