1978

**ELLPACK Network User's Guide**

Ron Boisvert

John Brophy

John R. Rice

*Purdue University, jrr@cs.purdue.edu*

John Warner

**Report Number:**

79-319
This report describes how to use the ELLPACK system through a set of
terminal command procedures, called macros, designed for easy, on-line
use. They allow users at remote sites to create an ELLPACK program for
solving a PDE, run an ELLPACK program, check the program output, have
results mailed, display data from the ELLPACK performance evaluation system,
send messages (trouble reports) to network members and provide on-line
aid in using the macros. Information about accessing the network and
Purdue's computer system is given elsewhere.
1. **SUMMARY OF CAPABILITIES**

The ELLPACK Network is designed to provide access to the ELLPACK system for a number of active members of the cooperative group. Many of these people have no experience with the Purdue operating system nor any desire to learn how to use it. The set of special "macros" described here are designed to allow one to use the ELLPACK system from a terminal with very little knowledge of the Purdue operating system. One exception to this is that the Purdue editor is used for the creation and maintenance of programs. This appears reasonable because (a) the Purdue editor is similar to many other editors and (b) it is a lot of work to produce an editor with even minimal capabilities.

The steps in a simple ELLPACK terminal session are:

**LOG ON.** This connects you to the Purdue system and initiates the ELLPACK system. Basically you give your account number, identification and password plus signal that you want to use the ELLPACK macros.

**CREATE A PROGRAM FILE.** You use the editor to write your program (and correct errors). When you finish you have a program (a file) with a name (for example, PROGA).

**RUN A JOB.** The RUN macro submits a job to run by the statement RUN, Program Name (for example, RUN, PROGA) This job is automatically assigned a name (for example R6) which is printed out. One may make several jobs from one program by changing parameters (or fixing errors).
CHECK THE PROGRAMMING OUTPUT. The normal output from an ELLPACK run is much too lengthy to receive at a terminal, yet one must see something. The macro LOOK does a number of things to help here; it is invoked by LOOK, program Name (for example LOOK, R6). This macro
(a) Monitors the execution of the job. Every 8 seconds a line is printed showing the most recently executed operating system step. This may not mean anything to you, but it shows that something is happening
(b) Returns a message saying whether the run was successful or not. If the run failed then the macro
(c) Searches the output for errors and prints the line with the error and the associated error message. The errors detected are:
1. ELLPACK Language errors found by the preprocessor.
2. Fortran errors in compiling the control program.
3. Memory overflow in loading the program.
4. Unsatisfied externals (missing functions or subroutines) found in loading the program.
5. Arithmetic errors in execution.
6. Abnormal terminations.
7. Excessive use of time, I/O units, memory, disk tracks, printer output, etc.
8. Use of non-existent files
(d) Displays the values of the MAX verbs (MAX-SOLUTION, MAX-RESIDUAL, MAX-TRUE or MAX-ERROR) if the run worked and you want it.
When these steps are complete, the output file may be made a "local" file for further examination using the editor. In most cases the source of any errors can be identified and fixed without any detailed knowledge of the internal operation of ELLPACK or the operating system.

MAIL COMPLETE OUTPUT. You may have the complete output mailed to you by saying MAIL, job name (for example, MAIL, R6). All network users have their addresses in the system and the printed output will be sent by first class mail not long after the run.

SEND AND RECEIVE MESSAGES. Any user can send messages to any other or report troubles to "ELLPACK" using the MESS macro. The recipient's ID is used as address and a list of ID's with user's names is displayed by MESS, USERS.

DISPLAY PERFORMANCE EVALUATION DATA. The macro GETDATA leads one through the steps to obtain data from the ELLPACK data base on PDE software performance evaluation. You must be familiar with the "problem-parameter set numbers" of the PDE population and with the encoding of methods (for example, 1/14/20/ is 5-POINT STAR/NATURAL/LINPACK BAND). One may also obtain some statistical analysis and plots of sets of data.

ON-LINE AID IN USING MACROS. The macro AID lists the other macros of the set. The statement AID, macro name (for example AID, LOOK) gives a one sentence definition of the macro plus its syntax.

The very nature of this remote access to ELLPACK requires that it be terminal, on-line oriented. However, ELLPACK is not a natural candidate for on-line use because many ELLPACK jobs take a long time to run. The
common causes of delays are

A. Solving the PDE. This can take a fair number of seconds and a large amount of memory. Thus program roll-outs are likely which delay the completion of the job even more.

B. First access to ELLPACK. The ELLPACK data base on performance evaluation is "tape stages" which means the first user each day must wait for a tape to be mounted and copied to disk. This action is automatic, but not fast (a 5-10 minute delay is typical)

C. System Overloading. As the number of users grows, the system response slows down for all activities. Over 100 users slows down simple items noticeable. The best times to use the system are: (a) right after it starts up (9:00 A.M., EST), (b) late at night (11:00 P.M. to 5:00 A.M.), (c) Purdue vacation periods.

Finally, we note that you can use any of the Purdue operating system commands during an ELLPACK session; this report explains only those special or absolutely necessary to running ELLPACK.

It is not necessary to wait at the terminal for your job to finish. If you are both logged off and have not MAILED the job, the output will be saved on a disk file until the end of the day. When you log back on you may use the macros described later to view the output as if you had not logged off. Note, that if you do not retrieve your output before the end of the day, it will be lost. This feature cannot be guaranteed because on days with heavy usage, the system may delete the file containing the output.

2. ANOTATED EXAMPLE SESSION

The following listing is a cut down version of an actual terminal session to run an ELLPACK program. The original program has errors of various types and the messages received and actions taken can be seen. The listing also shows the basic mechanics of logging on, creating and editing a file.
ELLPACK PREPROCESSOR ERRORS:

135.000=  Y=1,  U=3.
136.000=  ***  ***  FATAL ERROR  =  SIGN MISSING FOR BOUNDARY VARIABLE IDENTIFICATION,
137.000=  ***  ***  FATAL ERROR  TOO FEW BOUNDARIES FOR X
138.000=  3CONTINUE (This is a bug in the preprocessor, wrong line is printed)
139.000=  ***  ***  FATAL ERROR  NO = SIGN IN BOUNDARY CONDITION

CONTROL PROGRAM COMPILATION ERRORS:

21.30.07.  2  FATAL ERRORS IN ELL77
21.30.07.  2  FATAL ERRORS IN PDE
21.30.07.  2  FATAL ERRORS IN PDERHS
21.30.08.  2  FATAL ERRORS IN BCOND
122.  005303B  Bx = /
**(122)  -  FATAL------------------------------------------USE OF = BEFORE / OR MISSING OPERAND
**(122)  -  FATAL------------------------------------------EXPRESSION ENDING WITH /
9.  000005B  CVALUES( 7 ) $= 6.X#X#E<X+Y>3B<XY>+XY-8)
**(9)  -  WARNING  ------------------------------------------MISSING PERIOD AFTRE X
**(9)  -  FATAL------------------------------------------ILLEGAL CHARACTER IS IGNORED
**(9)  -  FATAL------------------------------------------MISSING ) OR EXTRA ( 
**(9)  -  FATAL------------------------------------------MISSING ( OR EXTRA )
2.  000000B  PDERHS $= 6.X#X#E<XY>3B<XY>+XY-8)
**(2)  -  WARNING  ------------------------------------------MISSING PERIOD AFTER X
**(2)  -  WARNING  ------------------------------------------ILLEGAL CHARACTER IS IGNORED
**(2)  -  FATAL------------------------------------------MISSING ( OR EXTRA )
**(2)  -  FATAL------------------------------------------MISSING ) OR EXTRA ( 
15.  000021B  GO TO 999
** (15)  -  FATAL------------------------------------------STATEMENT NUMBER IS ILLEGAL OR MISSING
23.  000032B  END
** (15)  -  FATAL------------------------------------------STATEMENT NUMBER 3 NEVER APPEARED

END OF ERRORS

DO YOU WISH TO USE THE EDITOR TO LOOK AT THE OUTPUT? N
DO YOU WISH TO MAKE THE NO JOB OUTPUT INTO A LOCAL FILE? N
+++EDIT PROB
*IC/9/* /
1.000=EDU  UXX$HYY =6.X#X#E<XY>3B<XY>+XY-8)
+C/9/*( Choose 9( To 9.
1.000=EDU  UXX$HYY =6.X#X#E<XY>3B<XY>+XY-8)
+C/9/*(/ Choose 9. To 1.
1.000=EDU  UXX$HYY =6.X#X#E<XY>3B<XY>+XY-8)
1.000=ER. UXX+UYY =6.X*Y*EX(X+Y)*(X*Y+X+Y-1.)
1.000=ER. UXX+UYY =6.X*Y*EX(X+Y)*(X*Y+X+Y-3.)
MARY/3/ 1.000=ER. UXX+UYY =6.X*Y*EX(X+Y)*(X*Y+X+Y-1.)
$S  **RUN PROG**  Run PROG again
2135102  PROG RUN AS HI
$LOOK HI
NAME OX  M SITE OP TYPE FL CPT/ IOU MESSAGE 12/19 21:35
H1 0250 1 PROC 1360 EX 2 15000 192/ 24836 FILES;ELLPK77,T=U.
H1 0250 1 PROC 1363 EX 2 85100 192/ 24316 LOAD,LOG,PRELIB,5NFS
H1 0250 1 PROC 1363 EX 2 55900 192/ 24004 COMPILING ELL77
21.35.30. CIC/H1 - ELLPACK-EXECUTION-FAILED. Job Failed
H1 0250 X PROC 3777 X

ELLPACK PREPROCESSOR ERRORS:

135.000=  Y=1., U=3.
136.000=  *** *** FATAL ERROR = SIGN MISSING FOR BOUNDARY VARIABLE IDENTIFICATION
137.000=  *** *** FATAL ERROR TOO FEW BOUNDARIES FOR X
138.000=  3CONTINUE  preprocessor log
139.000=  *** *** FATAL ERROR NO = SIGN IN BOUNDARY CONDITION

CONTROL PROGRAM COMPILATION ERRORS:

21.35.27.  2 FATAL ERRORS IN ELL77
21.35.27.  1 FATAL ERRORS IN PDE
21.35.27.  1 FATAL ERRORS IN PDERHS
21.35.27.  2 FATAL ERRORS IN BCOND
123.  005301B  BX = /

**122** - FATAL---------------------------------------------USE OF = BEFORE / OR MISSING OPERAND
**122** - FATAL---------------------------------------------EXPRESSION ENDING WITH /
9.  0000058  CVALUS( 7) = 6.X*Y*EX(X+Y)*(X*Y+X+Y-3.)
7.  0000058  CVALUS( 7) = 6.X*Y*EX(X+Y)*(X*Y+X+Y-3.)

**9** - WARNING-----------------------------------------------MISSING PERIOD AFTER X
**9** - FATAL-----------------------------------------------USE OF EOR BEFORE * OR MISSING OPERAND
2.  0000000  PDERHS = 6.X*Y*EX(X+Y)*(X*Y+X+Y-3.)
2.  0000000  PDERHS = 6.X*Y*EX(X+Y)*(X*Y+X+Y-3.)

**2** - FATAL-----------------------------------------------USE OF EOR BEFORE * OR MISSING OPERAND
15.  000021B  GO TO 999

**15** - FATAL-----------------------------------------------STATEMENT NUMBER IS ILLEGAL OR MISSING

23.  000032B  END
**23** - FATAL-----------------------------------------------STATEMENT NUMBER 3 NEVER APPEARED

END OF ERRORS
DO YOU WISH TO USE THE EDITOR TO LOOK AT THE OUTPUT? N

DO YOU WISH TO MAKE THE H2 JOB OUTPUT INTO A LOCAL FILE? N

+++ED PROG
$F/BOUND/
2.000=BOUNDX=0., Y=0.
$C/D/D. /
2.000=BOUNDX=0., Y=0.
$1C/6,5/*/
1.000=EQ. UXX**UYY =6, XX**EX(X+Y)*(X+Y+3.)
$S
+++RUN PROG
21:38:23 PROG RUN AS H2
+++LOOK H2
NAME OX M SITE OP TYPE FL CPT/ IOU MESSAGE 12/19 21:38
H2 0441 1 PROCs 1360 EX 2 15000. 192/ 24836 FILES; ELLPK77, T=R,
H2 0441 1 PROCs 1362 EX 2 34100 192/ 24484 LOAD+LGO+FRELIB7, MNF
H2 0441 1 PROCs 1363 EX 2 13100 192/ 24212 COPYBF+HEADER+ELLPGM

21.38.52. CIC/H2 - ELLPACK-EXECUTION-FAILED, Failed again

H2 0441 X PROCs 3777 PR

ELLPACK PREPROCESSOR ERRORS:
133.000= O CONTINUE (Same preprocessor bug)
134.000= *** *** FATAL ERROR UNIDENTIFIED BOUNDARY CONDITION

CONTROL PROGRAM COMPILATION ERRORS:
21.38.48. 1 FATAL ERRORS IN BCOND
25. 000031B END
** FATAL--------------------------------STATEMENT NUMBER 1 NEVER APPEARED

END OF ERRORS

DO YOU WISH TO USE THE EDITOR TO LOOK AT THE OUTPUT? N

DO YOU WISH TO MAKE THE H2 JOB OUTPUT INTO A LOCAL FILE? N
First boundary condition is nonsense, so we fix it.

$2 \quad 2.000 = \text{BOUND}, \quad X=0, \quad Y=0.$
$3 \quad 3.000 = \quad X=1, \quad UX=3.**$
$4 \quad 4.000 = \quad Y=0, \quad U=0.$
$2 \quad 2.000 = \text{BOUND}, \quad X=0, \quad U=3.**$

+++RUN PROG
21:41:45 PROG RUN AS H3
+++LOOK H3

NAME  QX  M  SITE  GP  TYPE  FL  CPT/ IOU MESSAGE 12/19 21:41
H3  0322 1 PROC 1130 EX 2 15000 192/ 24952.
H3  0322 1 PROC 1362 EX 2 46100 192/ 24420 LOADLG,PRELIB,NMF
H3  0322 1 PROC 1363 EX 2 13100 192/ 24184 COPYBF,FORT,ELLPGM,1
H3  0322 1 PROC 1366 EX 2 15000 192/ 23620,CX/EXEC/UTION)

21.42.15. CIC/H3 - ELLPACK-EXECUTION-COMPLETE. 

H3  0322 X PROC 3777 PR

**LOADER MESSAGES**

512.000 = UNSATISFIED EXTERNALS.
513.000 = EXTERNAL REFERENCES
514.000 =
515.000 =
516.000 = EX PDE 7247
517.000 =
518.000 =

H3 RAN TO COMPLETION

However, we still have errors.

DO YOU WISH TO SEE THE OUTPUT FROM THE MAX VERBS? Y
NO MAX VERB OUTPUT. But there is none.

DO YOU WISH TO USE THE EDITOR TO LOOK AT THE OUTPUT? N

DO YOU WISH TO MAKE THE H3 JOB OUTPUT INTO A LOCAL FILE? N
+++EDIT PROG
$IC/EX/EXP we fix EX
1.000=EQ. UX+IYY = 6.***Y*EXP(X+Y)*(X**Y+X+Y-3.)
PROG RUN AS H4

NAME  DK  SITE  DP  TYPE  FL  CPT/  IOU  MESSAGE  12/19  2143
H4  0250  1  PROCS  1360  EX  2  15000  192/  24864  REWIND,DATA,FORT,ELL
H4  0250  1  PROCS  1362  EX  2  60100  192/  24352  LOAD,LGO,PRELIB,MNF
H4  0250  1  PROCS  1363  EX  2  13100  192/  24208  COPYDF,DATA,ELLPGM
H4  0250  1  PROCS  1364  EX  2  55000  192/  23984  COMPILING ELL77
H4  0250  1  PROCS  1365  EX  2  55000  192/  23792  LOAD,ELLGO,BINARY,

21.44.42. CIC/H4 - ELLPACK-EXECUTION-COMPLETE.

H4  0250  X  PROCS  3777  PR
H4  RAN TO COMPLETION

Look  did not find any errors.

DO YOU WISH TO SEE THE OUTPUT FROM THE MAX VERBS? Y  Ask to see output
NO MAX VERB OUTPUT. There is none.

DO YOU WISH TO USE THE EDITOR TO LOOK AT THE OUTPUT? Y  use the editor to look at output

#F! PRINT to last C! means last line)
1.000= 21.43.53.
2.000= 21.43.53.*
3.000= 21.43.53.*

Many lines omitted here.
92.000= 21.44.34,LOAD,LGO,BINARY,LIBR,HLLIB,REFLIB,
93.000= 21.44.37,EXECUTE,INPUT,OUTPUT,PLOT,SAVE,SCRCH,
94.000= 21.44.37,CX  4.906 SEC., NL 35400 WORDS.
95.000= 21.44.37. STOP
96.000= 21.44.37.SET(R2=T)
97.000= 21.44.38. GPPLLOT,
98.000= 21.44.38. NO DATA ON PLOT FILE.
99.000= 21.44.38. CPU ABORT.
100.000= 21.44.38.PROCEED.
101.000= 21.44.3

Stop listing by CNTRL-B
Stop editor by #S.

DO YOU WISH TO MAKE THE H4 JOB OUTPUT INTO A LOCAL FILE? N
$P!
1.000=EQ. UXX+UYY =6.8*Y*EXP(X+Y)*(X*Y*X+Y-3.)
2.000=BOUND. X=0. , U=3.*Y
3.000= X=1. , UX=3.*X
4.000= Y=0. , U=0.
5.000= Y=1. , U=3.
6.000=GRID, UNIFORM X=5 , UNIFORM Y = 4
7.000=DIS. HODIE-HELMHOLTZ
8.000=INDEX, NATURAL
9.000=SDL. YALE SPARSE
10.000=OUTPUT, MAX-SOLUTION
11.000=END.

on line 6 change 4 to 5 (Remember HODIE-HELMHOLTZ)

$S
$RUN PROG
$RUN AS HS

NAME  OX NM SITE OP. TYPE FL CPT/ IDU MESSAGE 12/19 21:49
HS 0167 1 PROCS 1130 IN 2 40000 104/ 25000
HS 0167 2 PROCS 1131 IN 2 40000 184/ 25000
HS 0167 1 PROCS 6777 RO 2 15000 192/ 23680 TRMSG.ELLPACK-EXECUT

21:51:27. CIC/HS — ELLPACK-EXECUTION-COMPLETE.

HS 0167 X PROCS 3777 PR
HS RAN TO COMPLETION  LOOK FOUND no errors.

DO YOU WISH TO SEE THE OUTPUT FROM THE MAX VERBS? y Ask to see output

NO MAX VERB OUTPUT. There is no output.

DO YOU WISH TO USE THE EDITOR TO LOOK AT THE OUTPUT? y Use Editor on the output

517.000=EOF Go To bottom of file
$-EOF
497.000=
498.000=
499.000=
500.000=
501.000=
502.000=
503.000=

DOMAIN = RECTANGLE ( 0, 1.0000E+00) X ( 0, 1.0000E+00)

DISCRETIZATION = UNIFORM 5 X 5 HX= 2.5000E-01 HY= 2.5000E-01

OUTPUT LEVEL = 1
DO YOU WISH TO MAKE THE H5 JOB OUTPUT INTO A LOCAL FILE? N
+EDIT PROG
   #F/UX=/
   3.000= pind UX=
   X=1. , UX=3.8X
   $C/UX/U change UX to U
   3.000= X=1.78 , U=3.8X
   $H
+RUN PROG 'SI'
   Run PROG as a sized (small) job.
21:54:154 PROG RUN AS H6
++LOOK
   NAME  DX  M SITE  GP  TYPE  FL  'CPT/ IOU MESSAGE 12/19 21:54
   H6   0417 1 PROC 2305 EX' 1 15000 40/ 2952 .
   H6   0417 1 PROC 2305 EX '1 15000 40/ 2548 LOAD+LOD+PRELIB7+MNF
   H6   0417 1 PROC 2311 EX' 1 60100 40/ 2344 LOAD+LOD+PRELIB7+MNF
   H6   0417 1 PROC 2314 EX 1 52000 40/ 2236 T+OUTPUT+PNCH+DATA+F
   H6   0417 1 PROC 2322 EX 1 55000 40/ 1896 LOAD+ELLGO+BINARY7+M

21:55:36. CIC/H6 - ELLPACK-EXECUTION-COMPLETE.

H6   0417 X PROC 3777 PR
H6 RAN TO COMPLETION

DO YOU WISH TO SEE THE OUTPUT FROM THE MAX VERBS? Y

MAX (ABS(SOLUTION)) ON 5 X 5 X 1 GRID = 3.18313219E100

END OF MAX VERBS OUTPUT.
DO YOU WISH TO USE THE EDITOR TO LOOK AT THE OUTPUT? N

DO YOU WISH TO MAKE THE H6 JOB OUTPUT INTO A LOCAL FILE? N

MAIL

How can we have the job MAILED?

MAIL ROUTES THE OUTPUT TO STANDARD BIN FOR MAILING.

SYNTAX: MAIL, JOBNAME1, JOBNAME2...[JOBNAME10]

++MAIL H6

MAIL JOB.

JOB "H6" MAILED

++LOG

log off

TCP L210 21:57:02, 12/19/79.
ESTIMATED SESSION COST $ 2.77
PLEASE TURN OFF TERMINAL. TNX.
3. LOGGING ON THE PURDUE SYSTEM

After the phone connection has been established, typing the "control" key followed by the B key (CTRL B) will signal your presence to the Purdue system. You then will be prompted for a sequence of identifications.

<table>
<thead>
<tr>
<th>Prompt</th>
<th>You Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCOUNT?</td>
<td>Your 5 digit account number.</td>
</tr>
<tr>
<td>ID?</td>
<td>Your 3 character identifier.</td>
</tr>
<tr>
<td>PASSWORD?</td>
<td>Your password.</td>
</tr>
<tr>
<td>SYSTEM?</td>
<td>&quot;S&quot;</td>
</tr>
</tbody>
</table>

Typing "S" signals that you are using the ELLPACK Network. You will then receive a short list of news items broadcast to all users of the Purdue system, a count of the number of active terminal users, and a listing of any messages sent to you by other members of the ELLPACK Network. You may also be asked if you intend to use the data base. A prompt of "+++" signals that the system is ready to receive your next command.

The following keys have special meaning during your terminal session:

<table>
<thead>
<tr>
<th>KEY STROKE</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTRL B</td>
<td>Interrupt. Immediately halts processing of current command.</td>
</tr>
<tr>
<td>ESCAPE</td>
<td>Pause. Temporarily halts text currently being printed at the terminal. Printing resumes when ESCAPE is hit again.</td>
</tr>
<tr>
<td>CTRL H</td>
<td>Backspace. Erases the last character typed on the current line.</td>
</tr>
<tr>
<td>RUBOUT</td>
<td>Erase line. Causes a backspace to the beginning of the current line.</td>
</tr>
</tbody>
</table>

To log off the system, simply type "LOG" in response to a "+++" prompt.
4. USING THE EDITOR TO CREATE OR FIX A PROGRAM

To solve a PDE problem on the ELLPACK Network it is necessary to create a file at the terminal, called a local file, that contains an ELLPACK input program describing the problem. (See the ELLPACK User's Guide for details of the ELLPACK language). To create a local file, type

```
CREATE, filename
```

where filename is the unique name you choose for this file. You will be prompted by a line number and the character "=" which signals that you are in input mode. You may now type in your ELLPACK program to signal the end of input, type

```
$STOP
```
in response to a "=" prompt.

Undoubtedly, errors were made while typing the ELLPACK program. To modify (or simply explore) a local file, type

```
EDIT, filename
```

You will now be "inside" the system editor, which will prompt you for editor commands with the character "#". When finished with the editor, type "STOP" in response to an editor prompt.

While in the editor there is always a current line number. It always starts at the number of the first line in the file (usually 1) and then is always the last line referred to by an editor command. Most editor commands take the form

```
n X parameters
```
where X is the command name and n is the first line number to which the
command applies. If n is omitted the current line is used. A partial
listing of editor commands follows.

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>n P m</td>
<td>Prints m lines starting at line n.</td>
</tr>
<tr>
<td>T</td>
<td>Repositions current line pointer to the top of the file.</td>
</tr>
<tr>
<td>B</td>
<td>Repositions current line pointer to the bottom of the file.</td>
</tr>
<tr>
<td>n N m</td>
<td>Prints the m lines after line n.</td>
</tr>
<tr>
<td>n D m</td>
<td>Deletes m lines starting with line n.</td>
</tr>
<tr>
<td>n X</td>
<td>Insert lines after line n. This switches you to input mode (when done, type &quot;#&quot; as a response to an &quot;=&quot; prompt).</td>
</tr>
<tr>
<td>n R</td>
<td>Replace line n. This switches you to input mode (see 1).</td>
</tr>
<tr>
<td>nC/old/new/</td>
<td>Changes the first occurrence of the string &quot;old&quot; to the string &quot;new&quot; on line n. (Note—any other character may be used in place of / (except blank)).</td>
</tr>
<tr>
<td>nF/text/</td>
<td>Find the first line following line n containing the string &quot;text&quot;.</td>
</tr>
</tbody>
</table>

To save a local file for later use, type

PUT, filename.

in response to a "+++" prompt. This file may be retrieved (i.e. made local)) at a later session by typing.

GET, filename.

To delete a saved file, type

DELETE, filename.

A saved file that remains unused for 14 days is automatically deleted by the system. Finally, to get a list of your saved files, type

INDEX.
For more information on these operations see computing center documents [2], LO-DEC and [3], LO-PIRATE.

5. RUNNING ELLPACK PROGRAMS

Once an ELLPACK program has been made into a local file, you are ready to execute it. To do this, type

```
RUN, filename
```

in response to a "+++" prompt, where filename is the name of the local file containing the ELLPACK program. The RUN macro creates a job containing your ELLPACK program and the control cards necessary to execute it and submits it to the batch system. The response from the system will be

```
XX.XX.XX filename RUN AS jobname
```

where XX.XX.XX is the current time and jobname is the system generated name for this job. To see the output from this job you must use the LOOK or MAIL commands.

The RUN macro has two other forms

```
RUN, filename, 'jobcard parameters', 'ELLPACK parameters'
RUN, filename, filename
```

In the first form one optionally specifies a list of jobcard parameters (described below) and/or ELLPACK system parameters (see [1] XEQ-ELLPACK). In the second form one specifies a second local file that contains all the control cards necessary for the execution of the ELLPACK program. This option is provided for those who are familiar with the Purdue operating system.

**Jobcard parameters**

This is a list of designators separated by commas that tell the system what types of resources are required by your job. The default set of
parameters for ELLPACK programs is the job size 'S2'. This says that
your job is a size 2 job. Job sizes run from 1 to 4 with 4 the largest.
If your job exceeds the size limitations of S2 (you will receive a message
of the form 'resource LIMIT'), simply submit with Sn, where n is greater
than your last run. Big jobs run slower, so there is a penalty for over-
estimating.

A second parameter of interest is the line limit, which defaults to
1000 printed lines of output. To specify more lines, use Ln, where n is
the number of lines.

For details on other jobcard parameters see the computer center docu-
ment [4], LO-CONTROL.
6. CHECKING THE PROGRAM OUTPUT

Once an ELLPACK job has been submitted, you may get a report on how the job fared by using the LOOK macro. This is done by typing

```
LOOK, jobname
```

in response to a "+++" prompt. If the job has not yet completed execution, the macro will display a line showing the job's status every eight seconds until it is done. If you do not wish to wait, you may hit "CTRL B" any time and the macro will return a "+++" prompt and you may then enter any other command. When your job finishes, your terminal session will be interrupted with one of the messages

```
XX.XX.XX. wd/jobname - ELLPACK-EXECUTION-COMPLETE
XX.XX.XX. wd/jobname - ELLPACK-EXECUTION-FAILED
```

where XX.XX.XX is the time, wd is your 3 character ID. This signals that the LOOK macro may now be run without delay.

LOOK scans the output of the job for error messages generated by the ELLPACK Preprocessor, the Minnesota FORTRAN Compiler (MNF), the system loader, and the system runtime monitor. The types of error messages are summarized in Table 1.

<table>
<thead>
<tr>
<th>ERROR TYPE</th>
<th>CAUSE/APPROPRIATE ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELLPACK PREPROCESSOR ERRORS</td>
<td>An ELLPACK language syntax error was found. The message indicates the nature of the error, and the line where the error was detected is listed. You should EDIT the local file containing the ELLPACK program to correct the mistake.</td>
</tr>
</tbody>
</table>
### Control Program

#### Compilation Errors

These are errors in the compilation of the FORTRAN program generated by the Preprocessor. The most frequent cause of error here is a syntax error in an arithmetic expression in either the EQUATION, DOMAIN, BOUNDARY or FORTRAN segments of the ELLPACK program. The message indicates the nature of the error and the line where the error was detected is listed. You should EDIT the local file containing the ELLPACK program to correct the mistake.

#### Loader Messages

<table>
<thead>
<tr>
<th>Description</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) Memory overflow in processing load</td>
<td>Your estimate of the job size was too small. The amount of memory (FL) required to load the program will be listed (in octal). If this is greater than 150000 words, then you have exceeded the capacity of the machine. You should EDIT the ELLPACK program, reducing the size of the grid and/or the number of modules used. If the required memory is less than 150000, then rerun your program using <code>RUN, filename, 'MFn'</code> where <code>n</code> is the required memory. You have used a non-standard FORTRAN function name in the ELLPACK program, and failed to provide the function in the FORTRAN segment. This also occurs when a function names are misspelled. A list of the unsatisfied external reference names will be printed. You should EDIT the ELLPACK program to make the required changes.</td>
</tr>
<tr>
<td>(b) Unsatisfied External References</td>
<td></td>
</tr>
</tbody>
</table>

#### Execution Errors

<table>
<thead>
<tr>
<th>Description</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) Resource limits:</td>
<td>Your job size estimate was too small. The job should be RUN again with a larger job size parameter (see the description of RUN).</td>
</tr>
<tr>
<td>I/O UNIT LIMIT or TRACK COUNT LIMIT</td>
<td></td>
</tr>
<tr>
<td>LINE LIMIT or TIME LIMIT</td>
<td></td>
</tr>
<tr>
<td>(b) Arithmetic Error MODE = k</td>
<td>k = 1: address out of range&lt;br&gt;k = 2: infinite operand&lt;br&gt;k = 4: indefinite operand (i.e. O./O.)&lt;br&gt;These errors usually occur as secondary effects of other errors. You should scan the output file for error messages printed by ELLPACK modules (see below). If no such messages can be found, a bug may exist in a module. A trouble report should be filed (see MESS).</td>
</tr>
</tbody>
</table>
(c) Other errors

A number of other errors detected at runtime are found by this macro. If the message is not self-explanatory you should proceed as in (b) above.

If no errors were found, the LOOK macro prints the message "jobname RAN TO COMPLETION". It is important to realize that this does not imply that the desired output was generated. It means only that the ELLPACK control program terminated normally (it executed a STOP statement). Any ELLPACK module may have stopped the program prematurely upon discovering an error in its interface (This typically occurs when a user specifies a problem not in the domain of the module sequence). Although the module may have printed an error message, the LOOK macro does not currently recognize these.

Before the LOOK macro ends you will be asked if you wish to use the editor to view the job's output. After you leave the editor you will be asked if you want to make the output a local file whose name is the same as the jobname. (There is a limit to the number of local files you may have at a time. To list the names of your local files, type FILES, to delete a local file, type RETURN filename.)

Finally, LOOK will ask whether you wish to see the output of any MAX verbs specified in the ELLPACK input program. It is usually a good idea to check these, since they provide the simplest means of checking whether the answers produced were reasonable.

7. HAVING RESULTS MAILED

To have the output from a job mailed to you, type

```
MAIL, jobname
```

in response to a "+++" prompt. After a job is MAIled, it can no longer be LOOKed at.
8. **DISPLAYING AND ANALYZING PERFORMANCE EVALUATION DATA**

Substantial amounts of data on the performance of ELLPACK modules on a standard set of problems has been generated at Purdue and this data is available for inspection by Network users via the GETDATA macro. For each problem-method pair the performance evaluation data base contains a table of information, each line of the table representing the solution of the given problem by the given method on a single grid. The data available on each run is described in the following table.

<table>
<thead>
<tr>
<th>Entry name</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATE</td>
<td>Date of the test run</td>
</tr>
<tr>
<td>TIME</td>
<td>Time of the test run</td>
</tr>
<tr>
<td>NX</td>
<td>Number of grid lines in x-direction</td>
</tr>
<tr>
<td>NY</td>
<td>Number of grid lines in y-direction</td>
</tr>
<tr>
<td>HMAX</td>
<td>Largest grid spacing in any direction</td>
</tr>
<tr>
<td>NUMBEQ</td>
<td>Number of unknowns in linear system</td>
</tr>
<tr>
<td>ERRMAX</td>
<td>Max error on grid points</td>
</tr>
<tr>
<td>ERRMAXF</td>
<td>Max error on a fixed 20 x 20 grid</td>
</tr>
<tr>
<td>ERRL2</td>
<td>Discrete-L2 error at grid points</td>
</tr>
<tr>
<td>RESMAX</td>
<td>Max residual* at midpoints of grid rectangles</td>
</tr>
<tr>
<td>RESMXR</td>
<td>Max (residual/right side) at midpoints of grid rectangles</td>
</tr>
<tr>
<td>RESL2</td>
<td>Discrete L2 residual at midpoints of grid rectangles</td>
</tr>
<tr>
<td>SOLMAX</td>
<td>Max absolute value of solution on grid.</td>
</tr>
<tr>
<td>NIT</td>
<td>Number of iterations for ITPACK modules</td>
</tr>
<tr>
<td>MEM</td>
<td>Memory estimate (words)</td>
</tr>
<tr>
<td>TOTLT</td>
<td>Total execution time (sec.)</td>
</tr>
<tr>
<td>TIME1</td>
<td>Discretization module time (sec.)</td>
</tr>
<tr>
<td>TIME2</td>
<td>Indexing module time (sec)</td>
</tr>
<tr>
<td>TIME3</td>
<td>Solution module time (sec)</td>
</tr>
</tbody>
</table>

*This is the residual of the PDE operator: |L(computed solution) - right side|.
All data was generated on Purdue’s CDC 6500 system. The CDC 6000 series machines have a 60 bit word with 14.3 decimal places of precision in single precision arithmetic. All codes are compiled with the MNF FORTRAN compiler.

To gain access to this data, type

```
GETDATA
```

in response to a "+++" prompt. The macro will prompt you for a list of problem names (problem number and parameter set number, at most 60 pairs) as well as methods (at most 20). Method names are described in the Appendix. The PDE population is described in [7]. Four options for processing the data for all problem-method pairs specified are available:

1. Display the raw data at the terminal.
2. Have a listing of the raw data mailed.
3. Analyze plot components of the data.
4. Have analysis/plots mailed.

You will be prompted for each of these items. The first two are self-explanatory. The third option allows the analysis of one performance indicator with respect to one measure of problem size. The possible choices are

**Performance Indicators (y-axis)**

- ERR-NODES (= ERRMAX/SOLMAX)
- ERR-GRID (= ERRMAXF/SOLMAX)
- ERRL2
- RESMAX
- RESL2
- RESM2ER
Measures of Problem Size (x-axis)

NUMBER
MEMORY
TLTIME
TIME 1
TIME 2
TIME 3

The analysis program will compute basic statistics based upon linear least squares fits to the data for each problem-method pair. Both means, medians and extremes of the slope of the least squares lines over the chosen problem set are computed. For ERR-NODES and ERR-GRID, the mean resource levels required to attain 5%, 0.5% and 0.05% accuracy are also computed. The methods are ranked based upon each performance indicator and the Friedman, Kendall and Babington-Smith test is computed to indicate differences in average rank significant at the 90% and 99% confidence levels, see reference [5] for more information.

You will also be asked if the data should be plotted. If so, one plot per problem will be generated with x and y axes as specified above. The data points for each method will be connected by a least squares line. The plot is on a log-log scale. No plots on the Purdue system are allowed to exceed 20 ft. in length, hence no more than 20 problems can be specified if plotting is to be done.

An example run of GETDATA follows with some comments.
TCB L070 11.00.58, 03/18/80, FULL DUPLEX
ACCOUNT?  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £
PASSWORD?  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £  £
THIS USERID LAST LOGGED OFF AT 19.06.53, 03/17/80.
SYSTEM? 13  Need to set up things correctly
PIRATE
164 ACTIVE USERS (MAX - 173)
DO YOU INTEND TO USE THE DATABASE TODAY? [Y]  Asks for database
NO MESSAGES
+++GETDATA
ENTER PROBLEM - PARAMETER SET NUMBERS (ONE PAIR PER LINE) HERE.
TYPE "STOP" WHEN FINISHED.
1.000=3-1  Problem 3 - Parameter Set 1
2.000=5-2  Problem 5 - Parameter Set 2
3.000=5  
ENTER METHOD NAMES (ONE PER LINE) HERE.
TYPE "STOP" WHEN FINISHED.
1.000=1/14/20/  5-point star / natural indexing / Linpack solution /
2.000=3/14/20/  collocation / natural indexing / Linpack solution /
3.000=5  
DO YOU WISH TO HAVE THE DATA RETRIEVED
FOR DISPLAY AT THE TERMINAL OR FOR LATER MAILING? [Y]

DO YOU WISH TO HAVE THE DATA ANALYZED? [Y]
The keywords available are:
ERR-NODES  ERR-GRID  ERRL2
RESMAX  RESL2  RESMXR
NUMBEQ  MEMORY  TLTIME
TIME1  TIME2  TIME3
NX  NY  HMAX

Y-AXIS KEYWORD?
? ERR-GRID

X-AXIS KEYWORD?
? TLTIME

DO YOU WISH TO HAVE THE DATA PLOTTED? [Y]
JOB T1 SENT TO RETRIEVE DATA.
JOB T2 SENT TO ANALYZE DATA.
MESSAGE(S) WILL BE TRANSMITTED TO YOUR TERMINAL WHEN
THE JOB(S) ARE COMPLETED. IF YOU WISH TO SEE THE
OUTPUT DISPLAYED AT THE TERMINAL, TYPE "SHOWDATA T1+T2"
(Do not "MAIL" the output until you are finished viewing it)
+++
### Data Retrieval Complete

11:57:43, CIC/T1 - DATA-RETRIEVAL-COMPLETE.

### Data Analysis Complete

12:09:21, CIC/T2 - DATA-ANALYSIS-COMPLETE.

### T1 Output Requested

T1 OUTPUT REQUESTED IS:

PROBLEM = 3
PARAMETER SET = 1 (A=1.5)
PROBLEM INFO = 22002100200222 000.38 090.60 000.00 070.40

**Method = 1/14/20/**

<table>
<thead>
<tr>
<th>DATE</th>
<th>TIME</th>
<th>NX</th>
<th>NY</th>
<th>HMAX</th>
<th>NRUNK</th>
<th>ERRMAX</th>
<th>ERRMFF</th>
<th>ERRL2</th>
</tr>
</thead>
<tbody>
<tr>
<td>10/22/78</td>
<td>17.46</td>
<td>5</td>
<td>5</td>
<td>2.50E-01</td>
<td>9.11E-03</td>
<td>1.5E-03</td>
<td>9.4E-05</td>
<td></td>
</tr>
<tr>
<td>10/22/78</td>
<td>17.46</td>
<td>9</td>
<td>9</td>
<td>1.25E-01</td>
<td>4.94E-04</td>
<td>5.3E-04</td>
<td>2.7E-05</td>
<td></td>
</tr>
<tr>
<td>10/22/78</td>
<td>17.46</td>
<td>13</td>
<td>13</td>
<td>0.33E-02</td>
<td>1.21E-04</td>
<td>3.2E-04</td>
<td>1.2E-05</td>
<td></td>
</tr>
<tr>
<td>10/22/78</td>
<td>17.46</td>
<td>17</td>
<td>17</td>
<td>6.25E-02</td>
<td>2.25E-04</td>
<td>2.1E-04</td>
<td>6.4E-06</td>
<td></td>
</tr>
<tr>
<td>10/22/78</td>
<td>17.46</td>
<td>21</td>
<td>21</td>
<td>5.00E-02</td>
<td>3.61E-04</td>
<td>1.5E-04</td>
<td>3.9E-06</td>
<td></td>
</tr>
<tr>
<td>10/22/78</td>
<td>17.46</td>
<td>25</td>
<td>25</td>
<td>4.17E-02</td>
<td>5.29E-04</td>
<td>1.2E-04</td>
<td>2.6E-06</td>
<td></td>
</tr>
</tbody>
</table>

**Resmax Resmr Resl2 Solmax NIT Mem Tolt Time1 Time2 Time3**

<table>
<thead>
<tr>
<th>Resmax</th>
<th>Resmr</th>
<th>Resl2</th>
<th>Solmax</th>
<th>NIT</th>
<th>Mem</th>
<th>Tolt</th>
<th>Time1</th>
<th>Time2</th>
<th>Time3</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.5E-02</td>
<td>9.2E-01</td>
<td>1.6E-02</td>
<td>9.2E-01</td>
<td>3453</td>
<td>.04</td>
<td>.02</td>
<td>0</td>
<td>.02</td>
<td></td>
</tr>
<tr>
<td>1.1E-01</td>
<td>9.6E-01</td>
<td>6.3E-03</td>
<td>1.9E-02</td>
<td>6001</td>
<td>.31</td>
<td>.11</td>
<td>0</td>
<td>.20</td>
<td></td>
</tr>
<tr>
<td>1.4E-01</td>
<td>9.8E-01</td>
<td>3.8E-03</td>
<td>2.2E-02</td>
<td>11557</td>
<td>1.05</td>
<td>.26</td>
<td>0</td>
<td>.78</td>
<td></td>
</tr>
<tr>
<td>1.7E-01</td>
<td>9.8E-01</td>
<td>2.9E-03</td>
<td>2.1E-02</td>
<td>21273</td>
<td>2.54</td>
<td>.48</td>
<td>0</td>
<td>2.06</td>
<td></td>
</tr>
<tr>
<td>1.9E-01</td>
<td>9.9E-01</td>
<td>2.2E-03</td>
<td>2.2E-02</td>
<td>36301</td>
<td>5.13</td>
<td>.26</td>
<td>0</td>
<td>4.39</td>
<td></td>
</tr>
<tr>
<td>2.1E-01</td>
<td>9.9E-01</td>
<td>1.9E-03</td>
<td>2.2E-02</td>
<td>57793</td>
<td>9.44</td>
<td>1.10</td>
<td>0</td>
<td>8.34</td>
<td></td>
</tr>
</tbody>
</table>

**Method = 3/14/20/**

<table>
<thead>
<tr>
<th>DATE</th>
<th>TIME</th>
<th>NX</th>
<th>NY</th>
<th>HMAX</th>
<th>NRUNK</th>
<th>ERRMAX</th>
<th>ERRMFF</th>
<th>ERRL2</th>
</tr>
</thead>
<tbody>
<tr>
<td>10/22/78</td>
<td>18.58</td>
<td>3</td>
<td>3</td>
<td>5.00E-01</td>
<td>16.5E-05</td>
<td>7.1E-04</td>
<td>5.7E-06</td>
<td></td>
</tr>
<tr>
<td>10/22/78</td>
<td>18.58</td>
<td>5</td>
<td>5</td>
<td>2.50E-01</td>
<td>64.5E-05</td>
<td>1.9E-04</td>
<td>4.3E-06</td>
<td></td>
</tr>
<tr>
<td>10/22/78</td>
<td>18.58</td>
<td>7</td>
<td>7</td>
<td>1.67E-01</td>
<td>144.3E-05</td>
<td>9.7E-05</td>
<td>2.3E-06</td>
<td></td>
</tr>
</tbody>
</table>

**Resmax Resmr Resl2 Solmax NIT Mem Tolt Time1 Time2 Time3**

<table>
<thead>
<tr>
<th>Resmax</th>
<th>Resmr</th>
<th>Resl2</th>
<th>Solmax</th>
<th>NIT</th>
<th>Mem</th>
<th>Tolt</th>
<th>Time1</th>
<th>Time2</th>
<th>Time3</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0E-02</td>
<td>1.9E-01</td>
<td>2.0E-02</td>
<td>2.1E-02</td>
<td>6235</td>
<td>.18</td>
<td>.09</td>
<td>0</td>
<td>.08</td>
<td></td>
</tr>
<tr>
<td>6.1E-02</td>
<td>1.8E-01</td>
<td>7.7E-03</td>
<td>2.1E-02</td>
<td>14911</td>
<td>1.01</td>
<td>.33</td>
<td>0</td>
<td>.68</td>
<td></td>
</tr>
<tr>
<td>6.5E-02</td>
<td>1.7E-01</td>
<td>4.8E-03</td>
<td>2.1E-02</td>
<td>31331</td>
<td>2.97</td>
<td>.72</td>
<td>0</td>
<td>2.25</td>
<td></td>
</tr>
</tbody>
</table>
T2 ANALYSIS OUTPUT REQUESTED IS:

PLOT
X=TLTIME
Y=ERR-GRID
PROBLEMS= 3, 1 / 5, 2 /
METHODS= 1/14/20/

<table>
<thead>
<tr>
<th>SLOPE ERR-GRID -TLTIME</th>
<th>AVE RANK</th>
<th>MINIMUM</th>
<th>1ST QUART</th>
<th>MEDIAN</th>
<th>3RD QUART</th>
<th>MAXIMUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/14/20/</td>
<td>2.00</td>
<td>-8.6E-01</td>
<td>-8.6E-01</td>
<td>-8.6E-01</td>
<td>-8.6E-01</td>
<td>4.7E-01</td>
</tr>
<tr>
<td>3/14/20/</td>
<td>1.00</td>
<td>-1.2E+00</td>
<td>-1.2E+00</td>
<td>-1.2E+00</td>
<td>-1.2E+00</td>
<td>-7.1E-01</td>
</tr>
<tr>
<td>5/00 ERR-GRID -TLTIME</td>
<td>AVE RANK</td>
<td>MINIMUM</td>
<td>1ST QUART</td>
<td>MEDIAN</td>
<td>3RD QUART</td>
<td>MAXIMUM</td>
</tr>
<tr>
<td>1/14/20/</td>
<td>1.00</td>
<td>7.9E-02</td>
<td>7.9E-02</td>
<td>7.9E-02</td>
<td>7.9E-02</td>
<td>9.8E-02</td>
</tr>
<tr>
<td>3/14/20/</td>
<td>2.00</td>
<td>9.8E-02</td>
<td>9.8E-02</td>
<td>9.8E-02</td>
<td>9.8E-02</td>
<td>1.2E-01</td>
</tr>
<tr>
<td>0.5 0/0 ERR-GRID -TLTIME</td>
<td>AVE RANK</td>
<td>MINIMUM</td>
<td>1ST QUART</td>
<td>MEDIAN</td>
<td>3RD QUART</td>
<td>MAXIMUM</td>
</tr>
<tr>
<td>1/14/20/</td>
<td>2.00</td>
<td>1.4E+00</td>
<td>1.4E+00</td>
<td>1.4E+00</td>
<td>1.4E+00</td>
<td>1.1E+01</td>
</tr>
<tr>
<td>3/14/20/</td>
<td>1.00</td>
<td>8.1E-01</td>
<td>8.1E-01</td>
<td>8.1E-01</td>
<td>8.1E-01</td>
<td>2.5E+00</td>
</tr>
</tbody>
</table>

AVE RANK DIFFERENCES SIGNIFICANT WITH 90 PERCENT CERTAINTY IF GREATER THAN 1.16

AVE RANK DIFFERENCES SIGNIFICANT WITH 99 PERCENT CERTAINTY IF GREATER THAN 1.82

++MAIL, T1 T2

These numbers tell you (more or less) whether you have enough data to reach statistically meaningful conclusions. As you can see, two problems will never be enough to get a significant difference in performance ranks.
9. **SENDING MESSAGES**

Users of the ELLPACK Network may send messages to each other or trouble reports to "ELLPACK" via the MESS macro. To do this, type

```
MESS, action
```

in response to a "+++" prompt,

where action is one of SEND, RECEIVE or USERS. (these may be abbreviated as S, R and U). The action taken are:

(a) MESS, R

All messages in your mailbox will be displayed at the terminal along with the name of the sender and the date and time sent. You will be asked if you want your messages saved; if you answer no, your mailbox will be reset to empty.

(b) MESS, S

You will be asked for your name and the three letter user ID of the person to receive the message. Finally, you will be prompted for each line of your message.

(c) MESS, U

Displays a list of each ELLPACK Network user along with his/her user ID.

To send a trouble report to "ELLPACK", type MESS, S and simply hit carriage return when asked for the user ID. If you are documenting an unexplainable error in an ELLPACK run, we suggest that you obtain the job output as a local file (via LOOK) and then save it (using PUT, jobname). If you tell us the jobname in your trouble report, we can then obtain the output to diagnose the error.
10. **AID: ONLINE AID IN USING THE MACROS**

Some minimal documentation on the ELLPACK macros is obtained by typing

```
AID, name
```

in response to a "+++" prompt. If the parameter "name" is omitted, a list of the ELLPACK macros is displayed. If "name" is an ELLPACK macro name, a brief description of the macro and its syntax will be displayed.

Detailed on-line documentation on many of the Purdue system terminal commands is available via the command

```
HELP, name
```

where "name" is a command. You may obtain documentation on GET, PUT, DELETE, INDEX, EDIT, and CREATE in this way. Other commands that you might find useful (but that we do not describe here) are

- **FILES** - list names of local files.
- **RETURN** - remove a local file.
- **PURGE** - remove job output from print queue
- **RELEASE** - halt execution of a job.
- **SEARCH** - check status of a job.
11. SAVED OUTPUT

Output is saved if both

(1) you are not logged on when the job finishes, and

(2) the job has not been MAILed.

A catalogue of outputs currently being saved can be obtained by typing CATOUT. For example, job R2 submitted by CIC would be identified as CICR2. If there was also a plot it would be saved as PCICR2. To mail the job type MAIL R2.

12. PFFILES

Any file stored in PFFILES not used for 15 days is deleted by the system. A list of your files and how many days left they have can be obtained by typing INDEX. To reset the number of days left to 15 type RESTDL. RESTDL will ask for your master password. If you have not set a master password, just hit a carriage return when asked.
13. GENERATING NEW DATA FOR THE DATABASE

To generate data for inclusion in the database, type GENDATA. It will prompt you for:

1. a list of problem-parameter set pairs (e.g., 12-2 for Problem 12, parameter set 2)
2. a method name, see appendix for formats
3. a list of grid sizes
4. another method name

If there are no more method names, just carriage return.

GENDATA then submits a job to generate the requested data and stores it in your PFILES under the name RUNLOG. To view the output use LOOK. To have the data entered into the database type SAVDATA after the job has finished. This will put a copy of the data in an ELLPACK account for later entry into the database. This new data cannot be viewed via GETDATA/SHOWDA until it is entered into the database.

14. DEFAULTS

The last file name used and the last jobname used, are remembered and are used as defaults where possible, with two exceptions:

1. after MAILing a job the last jobname is null, and
2. The job or jobs sent by GETDATA for later use by SHOWDA are remembered independently of the last jobname.
15. **ADDITIONAL MACROS**

This is needed only if you intend to write your own macros in addition to NETWORK macros.

The NETWORK macros parameter connections are:

- **G0-G10**: scratch
- **G11-G19**: available for user
- **G20-G22**: reserved for future use
- **G23**: data base tape number
- **G24**: data base name
- **G25**: prefix letter for jobname
- **G26**: GETDATA/SHOWDA jobnames
- **G27**: site for ROUTEd output
- **G28**: bin number for ROUTEd output
- **Z26**: last filename
- **Z27**: last jobname

These parameters are initialized by the macro `BEGIN` which should be called from your STARTUP macro.
APPENDIX: METHOD NAMES AND RECORD NUMBERS FOR PERFORMANCE DATA

Method names in the ELLPACK performance evaluation system are encoded as

discretization/indexing/solution/

where each component is a module number (a unique integer associated with each ELLPACK module), possibly followed by a list of parameters separated by commas. For example,

1/14/20/
9, IORDER=6///
5, METHOD=44, ISET=2, ETA=0.5/14/24/
5/14/24/

are four distinct methods. In the second example a module that combines all three components is specified. Note that "indexing" and "solution" are null in this case.

The ELLPACK module names and their associated numbers are given in the table below. Refer to the module description in the ELLPACK User's Guide [6] for the parameters associated with each module.
The modules available in ELLPACK 77 and their numbers are:

<table>
<thead>
<tr>
<th>Discretization</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. 5-POINT STAR</td>
<td>18. BAND SOLVE (Retired)</td>
</tr>
<tr>
<td>2. 7-POINT 3D</td>
<td>19. SYMMETRIC BAND (Retired)</td>
</tr>
<tr>
<td>3. PC-C1 COLLOCATION</td>
<td>20. LINPACK BAND</td>
</tr>
<tr>
<td>4. F3-C1 GALERKIN</td>
<td>21. LINPACK SPD BAND</td>
</tr>
<tr>
<td>5. HODIE-HELMHOLTZ</td>
<td>22. SPARSE GE-PIVOTING</td>
</tr>
<tr>
<td>6. HODIE-ACF</td>
<td>23-4. SOR (ITPACK-1)</td>
</tr>
<tr>
<td>7. HODIE-ACDEF</td>
<td>24. JACOBI SI (ITPACK-1)</td>
</tr>
<tr>
<td>8. HODIE 27-POINT 3-D</td>
<td>25. JACOBI CG (ITPACK-1)</td>
</tr>
<tr>
<td>9. FFT 9-POINT</td>
<td>26. SYMMETRIC SOR SI (ITPACK-1)</td>
</tr>
<tr>
<td>10. ZDEPESD</td>
<td>27. SYMMETRIC SOR CG (ITPACK-1)</td>
</tr>
<tr>
<td>11. MARCHING ALGORITHM</td>
<td>28. REDUCED SYSTEM SI (ITPACK-1)</td>
</tr>
<tr>
<td>12. DYAKANOV-CG</td>
<td>29. REDUCED SYSTEM CG (ITPACK-1)</td>
</tr>
<tr>
<td>13. DYAKANOV-CG 4</td>
<td>30. YALE SPARSE</td>
</tr>
<tr>
<td></td>
<td>31. YALE ENVELOPE</td>
</tr>
<tr>
<td></td>
<td>32. NESTED DISSECTION</td>
</tr>
<tr>
<td></td>
<td>33. SOR (ITPACK-2)</td>
</tr>
<tr>
<td></td>
<td>34. JACOBI SI (ITPACK-2)</td>
</tr>
<tr>
<td></td>
<td>35. JACOBI CG (ITPACK-2)</td>
</tr>
<tr>
<td></td>
<td>36. SYMMETRIC SOR SI (ITPACK-2)</td>
</tr>
<tr>
<td></td>
<td>37. SYMMETRIC SOR CG (ITPACK-2)</td>
</tr>
<tr>
<td></td>
<td>38. REDUCED SYSTEM SI (ITPACK-2)</td>
</tr>
<tr>
<td></td>
<td>39. REDUCED SYSTEM CG (ITPACK-2)</td>
</tr>
</tbody>
</table>
REFERENCES

[1] R. F. Boisvert, XEQ-ELLPACK, CSDTR XXX, Computer Science Department, Purdue University, August, 1979


[4] Purdue University Computing Center, A description of control cards and job processing, LO-CONTROL, March, 1979


Numbering system has been changed as of 1/80. New numbering system will be documented in early spring 1980.