

1979

## ELLPACK Network Documentation Preliminary Version

Ron Boisvert

John R. Rice  
*Purdue University, jrr@cs.purdue.edu*

John Warner

Report Number:  
79-308

---

Boisvert, Ron; Rice, John R.; and Warner, John, "ELLPACK Network Documentation Preliminary Version" (1979). *Department of Computer Science Technical Reports*. Paper 237.  
<https://docs.lib.purdue.edu/cstech/237>

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries.  
Please contact [epubs@purdue.edu](mailto:epubs@purdue.edu) for additional information.

**ELLPACK NETWORK DOCUMENTATION  
PRELIMINARY VERSION**

**Ron Boisvert  
John R. Rice  
John Warner**

**CSD TR #308  
July 1979**

ELLPACK NETWORK DOCUMENTATION  
PRELIMINARY VERSION  
RON BOISVERT, JOHN RICE and JOHN WARNER

July 15, 1979

CSD-TR 308

ABSTRACT

The report describes how to use the ELLPACK system through a set of terminal command procedures, called macros, designed for easy, on-line use. The macros as described are a preliminary set; an expanded set will be prepared once some experience is gained with this preliminary set. The capabilities of the preliminary set allow users at remote sites to create an ELLPACK program for solving a PDE, run a 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 to using the macros. Information about accessing the network and Purdue's computer system is given elsewhere.

CONTENTS

1. Summary of capabilities
2. Annotated example session
3. Logging on the Purdue system
4. Using the editor to create or fix a program
5. Running ELLPACK programs
6. Checking the program output
7. Having results mailed
8. Displaying and analyzing performance evaluation data
9. Sending messages
10. AID: on-line aid in using the macros
11. References

## ELLPACK NETWORK DOCUMENTATION

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
  - 9. Fatal ELLPACK execution errors.
- (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. The mailing of plotted output is not so reliable in the preliminary version.

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 SEEDATA leads one through the steps to obtain data from the ELLPACK data base on PDE software performance evaluation. You must be familiar with the "record 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 staged" 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. Copying big files. Each invocation of the SEEDATA macro requires that the entire data base be brought into position to be read. This takes a few minutes and the process can be further delayed by job roll-outs.
- D. System Overloading. As the number of users grows, the system response slows down for all activities. Over 100 users slows down simple items noticeably. The best times to use the system are:  
(a) right after it starts up (9:30 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.

## 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.

1

13:42:09 FROM RUN AS R1 } THIS IS THE STANDARD  
 LOG ON SEQUENCE  
 LARGE PARENT FILES WERE RELOADED FROM THEIR  
 BACKUP TAPES AT 23:00 SATURDAY 07/13/79. SMALL  
 PERMANENT FILES (FILES) WERE NOT AFFECTED.  
 MESSAGES FROM POCORR  
 COMPUTING CENTER CAN BE  
 SUPPRESSED BY TYPING CANCEL

THIS USERID LAST LOGGED OFF AT 14:12:05, 07/13/79,  
 BY 14:33. YOUR TERMINAL SESSION MUST BE TERMINATED.  
 SYSTEMY [ ] THIS IS SYNTAX TO GET THINGS  
 FURATE JUST NOT RETURN IN RESPONSE TO "NIRATE"  
 124 ACTIVE USERS (MAX = 140)

NO MESSAGES

+++CREATE, FROB

1.000=EG, UXX\*UY = 6.X\*YEX(X+Y)B(X\*Y+X+Y-8.)  
 2.000=ROUND, X+), U=),XCK  
 3.000=ROUNDX=0, Y=0,  
 4.000=X=1, U=3,\*X  
 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=SOL, YALE SPARSE  
 10.000=OUTPUT, MAX-SOLUTION  
 11.000=END,  
 12.000=IS ← THIS STOPS CREATE

CREATE  
 THE BACKLACK PROGRAM  
 PRAG

+++

+++RUN FROB

13:42:09 FROM RUN AS R1 } RUN THIS PROGRAM

+++LOOK,R1 START MONITORING ITS ALGORITHMS

R1 0266 1 PROCS 1455 RO 2 22100 192/ 9548 LOAD,LGO,PRELIB7,MNF  
 R1 0266 1 PROCS 1455 RO 2 22100 192/ 9548 LOAD,LGO,PRELIB7,MNF  
 R1 0266 1 PROCS 1455 RO 2 22100 192/ 9548 LOAD,LGO,PRELIB7,MNF  
 R1 0266 1 PROCS 1455 RO 2 22100 192/ 9548 LOAD,LGO,PRELIB7,MNF  
 LOOK IGNORED

NOTICED PROGRAMS STAYING WHILE WAITING SO HIT CANCEL  
 TO GET OUT OF LOOK AND FIX PROGRAM

+++EDIT FROB

13:42:09 FROM RUN AS R1 } CHANGE, BC Z \*9

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*9XX\*Y+X+Y-8.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-8.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

1.000=ER, UXX\*UY = 6.X\*YEX(X+Y)\*X\*Y+X+Y-3.)

+++LOOK,R2 LOOK TO SEE WHAT R2 IS DOING - NOT MUCH

R2 0267 1 PROCS 1455 RO 2 22100 192/ 9548 LOAD,LGO,PRELIB7,MNF  
 R2 0267 1 PROCS 1455 RO 2 22100 192/ 9548 LOAD,LGO,PRELIB7,MNF  
 R2 0267 1 PROCS 1455 RO 2 22100 192/ 9548 LOAD,LGO,PRELIB7,MNF  
 R2 0267 1 PROCS 1455 RO 2 22100 192/ 9548 LOAD,LGO,PRELIB7,MNF  
 LOOK IGNORED

MEANS JOB IS ROLLED OUT, PAK EXECUTING

```

+++LOOK R1
R1 0266 1 PROCS 1503 RO 2 13100 192/ 9244 COPYBF,HEADER,ELLPGM
R1 0266 1 PROCS 1503 RO 2 13100 192/ 9244 COPYBF,HEADER,ELLPGM
R1 0266 1 PROCS 1503 RO 2 13100 192/ 9244 COPYBF,HEADER,ELLPGM
R1 0266 1 PROCS 1503 RO 2 13100 192/ 9244 COPYBF,HEADER,ELLPGM
R1 0266 1 PROCS 1503 RO 2 13100 192/ 9244 COPYBF,HEADER,ELLPGM
R1 0266 1 PROCS 1503 RO 2 13100 192/ 9244 COPYBF,HEADER,ELLPGM
R1 0266 1 PROCS 1503 RO 2 13100 192/ 9244 COPYBF,HEADER,ELLPGM
R1 0266 1 PROCS 1503 RO 2 13100 192/ 9244 COPYBF,HEADER,ELLPGM
R1 0266 1 PROCS 1507 RO 2 13100 192/ 9244 COPYBF,HEADER,ELLPGM
LOOK IGNORED GIVE UP AND RESUBMIT JOB AS A SIZE 1 JOB INSTEAD OF THE
                DEFAULT SIZE 2

```

(2)

```

+++RUN,PROB(S1) = SIZE 1
13:52:59 PROG RUN AS R3

```

```

+++LOOK R3
R3 0025 2 PROCS 2316 RO 1 55000 40/ 2208 RFL,55000.
R3 0025 2 PROCS 2316 RO 1 55000 40/ 2208 RFL,55000.
R3 0025 2 PROCS 2316 EX 1 55000 40/ 2208 RFL,55000.
R3 0025 2 PROCS 5 RO 1 55000 40/ 1992 TRMSG,NA.ELLPACK-EXE
R3 0025 0 PROCS 5 RO 1 55000 40/ 1992 TRMSG,NA.ELLPACK-EXE

```

13.54.32. CIC/R3 - ELLPACK-EXECUTION-FAILED! WHOOPS, RUN FAILED

R3 0025 X PROCS 3777 PR WAIT HERE WHILE ERRORS ARE COLLECTED

ELLPACK PREPROCESSOR ERRORS:

```

84.000= Y=1, U=3.
85.000= *** ** FATAL ERROR = SIGN MISSING FOR BOUNDARY VARIABLE IDENTIFICATION ,
86.000= *** ** FATAL ERROR TOO FEW BOUNDARIES FOR X
87.000= 3CONTINUE (THIS IS A "BUG" IN THE PREPROCESSOR, WRONG LINE IS PRINTED)
88.000= *** ** FATAL ERROR NO = SIGN IN BOUNDARY CONDITION

```

CONTROL PROGRAM COMPILATION ERRORS:

```

48.000= 13.54.31. 2 FATAL ERRORS IN ELL77
49.000= 13.54.32. 1 FATAL ERRORS IN PDE
50.000= 13.54.32. 1 FATAL ERRORS IN PDERHS
51.000= 13.54.32. 2 FATAL ERRORS IN BCOND
267.000= 122. 005303B BX = /
268.000= **(122) - FATAL-----USE OF = BEFORE / OR MISSING OPERAND
269.000= **(122) - FATAL-----EXPRESSION ENDING WITH /
405.000= 9. 000005B CVALUS( 7) = 6.X*Y*EX(X+Y)*(X*Y+X+Y-3.)
406.000= (9) - WARNING-----MISSING PERIOD AFTER X
407.000= ** (9) - FATAL-----USE OF EOR BEFORE * OR MISSING OPERAND
412.000= 2. 000000B PDERHS = 6.X*Y*EX(X+Y)*(X*Y+X+Y-3.)
413.000= (2) - WARNING-----MISSING PERIOD AFTER X
414.000= ** (2) - FATAL-----USE OF EOR BEFORE * OR MISSING OPERAND
435.000= 15. 000021B GO TO 999
436.000= 3CONTINUE
438.000= ** (15) - FATAL-----STATEMENT NUMBER IS ILLEGAL OR MISSING
447.000= 23. 000032B END
448.000= ** FATAL-----STATEMENT NUMBER 3 NEVER APPEARED

```

BELOW IS OUTPUT FROM THE FORTRAN COMPILER, HOPEFULLY, YOU CAN SEE YOUR ERRORS HERE

END OF ERRORS

```

+++EDIT PROG
#6C/4/5/
6.000=GRID. UNIFORM X = 5 * UNIFORM Y = 4
7.000=DIS. HODIE-HELMHOLTZ
8.000=INDEX. NATURAL
9.000=SQL. YALE SPARSE
10.000=OUTPUT. MAX-SOLUTION
11.000=END.
12.000=#EOR

```

AND REMEMBER  
THAT HODIE-HELMHOLTZ  
NEEDS A UNIFORM MESH

5

```

+++EDIT PROG
#6C/4/5/
6.000=GRID. UNIFORM X = 5 * UNIFORM Y = 5
#S
+++RUN PROG 'S1'
14:16:21 PROG RUN AS R7
+++LOOK R7
R7 0022 2 PROCS 5 RO 1 36300 40/ 1816 TRMSG,NA.ELLPACK-EXE

```

14.18.06. CIC/R7 - ELLPACK-EXECUTION-SUCCESSFUL!

R7 0022 X PROCS 3777 PR

R7 RAN TO COMPLETION

DO YOU WISH TO SEE THE OUTPUT FROM THE MAX VERBS? Y  
NO MAX VERB OUTPUT!

AND THERE IS STILL  
NO OUTPUT

DO YOU WISH TO MAKE THE R7 JOB OUTPUT INTO A LOCAL FILE? Y

R7 IS NOW A LOCAL FILE!

SO WE START SEARCHING R7

```

+++EDIT R7
#F/END/
255.000= C END OF DEFAULT INITIALIZATIONS
#F/END/
382.000= 226. 005633B END
#F/END./
NOT FOUND
#382P5
382.000= 226. 005633B END
383.000= C +++++
384.000= C +++++
385.000= C +++++
386.000= C +++++ FILE FORTRAN FOR ELLPACK 77 +++++
#F/END/
NOT FOUND
#436P10
436.000= 28. 000036B END
437.000=
438.000=#EOR
439.000=1
440.000=
441.000=
442.000=

```

WE SKIP THROUGH THE FORTRAN  
PROGRAM BY JUMPING FROM  
"END" TO "END"

MEANS: AT LINE 382, PRINT, 5 LINES

BUT WE ARE STILL IN THE FORTRAN CONTROL PROGRAM

LATER WE FIND THE LAST END (IT SHOULD ALWAYS BE ABOUT LINE)  
430 TO 450

AND PRINT 10 LINES STARTING AT 436

FINITE DIFFERENCE MODULE - HODIE-HELMHOLTZ

11

444.000=  
445.000=  
\*P20  
445.000=  
446.000=  
447.000=  
448.000=  
449.000=  
450.000=  
451.000=  
452.000=  
453.000=  
454.000=  
TIONS.  
455.000=  
456.000=  
457.000=  
458.000=  
459.000=  
460.000=  
461.000=  
462.000=#EOR  
463.000=#EOF  
#S  
+++EDIT PROG  
#F/UX=  
3.000=  
#C/UX/U/  
3.000=  
#RUN PROG 'S1'  
SYNTAX ERROR  
#S  
+++RUN PROG 'S1'  
14125:03 PRDG RUN AS RB  
+++LOOK RB

PRINT 20  
MORE LINES

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

6

\*\*\*INPUT BOUNDARY CONDS ARE NEUMAN OR MIXED ----- HODIE-HELMHOLTZ ONLY HANDLES DIRICHLET BOUNDARY CONDI  
EXCEPTION -- NEUMANN PROBLEM FOR POISSON EQUATION

HERE IS THE TROUBLE, HODIE-HELMHOLTZ HANDLE MIXED B.C. ONLY FOR  
CONSTANT COEFFICIENT PROBLEMS - AND WE DIDNT SAY WE HAD  
SUCH A PROBLEM

FIND THE "UX" IN PROGRAM

CHANGE IT TO "U"

RUN AGAIN

#S  
+++EDIT PROG  
#F/UX=  
3.000= X=1. , UX=3.\*X  
#C/UX/U/  
3.000= X=1. , U=3.\*X  
#RUN PROG 'S1'  
SYNTAX ERROR  
#S  
+++RUN PROG 'S1'  
14125:03 PRDG RUN AS RB  
+++LOOK RB  
RB 0136 2 PROCS 2312 RO 1 53100 40/ 2412 LOAD,LGO,PRELIB7,MNF  
RB 0136 2 PROCS 2315 RO 1 65100 40/ 2348 LOAD,LGO,PRELIB7,MNF  
RB 0136 2 PROCS 2321 RO 1 55000 40/ 2208 RFL,55000.  
RB 0136 2 PROCS 2322 EX 1 55000 40/ 2208 RFL,55000.  
RB 0136 2 PROCS 5 RO 1 33600 40/ 1816 TRMSG,NA,ELLPACK-EXE  
RB 0136 2 PROCS 5 RO 1 33600 40/ 1816 TRMSG,NA,ELLPACK-EXE

14.26.17. CIC/RB - ELLPACK-EXECUTION-SUCCESSFUL!

RB 0136 X PROCS 3777 PR

RB RAN TO COMPLETION

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

OUTPUT OF MAX ABSOLUTE VALUE VERBS:

NOUN	GRID	VALUE
SOLUTION	5 5	3.18313219E+00

AND NOW WE HAVE OUTPUT

END OF MAX VERBS OUTPUT!

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

+++MAIL RB  
CICH9 = C1C0541 AT MATH

WE ASK <sup>PUR</sup> THE COMPLETE PRINTED OUTPUT  
TO BE MAILED TO US

JOB 'RB' MAILED

+++AID MAIL

WE JUST DOUBLE CHECK  
ON HOW TO USE MAIL

MAIL ROUTES THE OUTPUT TO STANDARD BIN FOR MAILING.

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

+++LOG

TCB L327 14.28.56. 07/16/79.  
ESTIMATED SESSION COST \$ 1.89  
PLEASE TURN OFF TERMINAL. TNX.

7

### 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.

Prompt	You Type
ACCOUNT?	Your 5 digit account number.
ID?	Your 3 character identifier.
PASSWORD?	Your password.
SYSTEM?	,S

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. A prompt of "+++" signals that the system is ready to receive your next command.

The following keys have special meaning during your terminal session:

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

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 as 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.

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

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-LEARNPEP 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 document [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 1  
Errors Found by LOOK

ERROR TYPE	CAUSE/APPROPRIATE ACTION
ELLPACK PREPROCESSOR ERRORS	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.

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

(a) Memory overflow  
in processing load

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

RUN, filename, 'MF<sub>n</sub>'

(b) Unsatisfied  
External  
References

where n 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.

EXECUTION ERRORS

(a) Resource limits:  
I/O UNIT LIMIT or  
TRACK COUNT LIMIT or  
LINE LIMIT or  
TIME LIMIT

Your job size estimate was too small. The job should be RUN again with a larger job size parameter (see the description of RUN).

(b) ARITHMETIC ERROR  
MODE = K

k = 1 : address out of range  
k = 2 : infinite operand  
k = 4 : indefinite operand (i.e. 0./0.)  
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).

(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 the job's output be made into a local file. If you reply yes, then a local file whose name is the same as the jobname will exist. This file may be EDITed, allowing you to scan portions of the file to check the results or look for additional error messages.

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. This macro should be executed only after the job has finished executing (otherwise the macro will wait until the job has completed).

## 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 SEEDATA 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.

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

\*This is the residual of the PDE operator:  $|L(\text{computed solution}) - \text{right side}|$ .

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

SEEDATA

in response to a "+++" prompt. The macro will prompt you for a list of problem names (integer record numbers, at most 60) as well as method (at most 20). Method names are described in the Appendix along with a table that associates record numbers with problem numbers in the PDE population [4]. 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 (= ERRMXF/SOLMAX)  
ERRL2  
RESMAX  
RESL2  
RESMXR

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 record numbers can be specified if plotting is to be done.

An example run of SEEDATA follows with some comments.

TER 1322 1  
USER HAS TOGGLED TRANSMISSION MODE  
DE 1 10.31.10. 07/20/79. FULL DUPLEX  
ACCOUNT# ~~XXXX~~ ~~XXXX~~ } GIVE ACCOUNT INFORMATION  
PASSWORD#

LARGE PERMANENT FILES WERE RELOADED FROM THEIR  
BACKUP TAPES AT 23:00 SATURDAY 07/14/79. SMALL  
PERMANENT FILES (PFILES) WERE NOT AFFECTED.

THIS USERID LAST LOGGED OFF AT 16.37.17. 07/19/79.  
BY 11:31. YOUR TERMINAL SESSION MUST BE TERMINATED.  
SYSTEM# VS  
PIRADI

120 ACTIVE USERS (MAX - 128)

NO MESSAGES

FFHSEEDATA

ENTER RECORD NUMBERS (ONE PER LINE) HERE.  
TYPE "#STOP" WHEN FINISHED.

1.000=22  
2.000=23  
3.000=#S  
*THERE ARE RECORD NUMBERS THAT ONE  
MUST BE ABLE TO GIVE TO ASK FOR  
PARTICULAR PROBLEMS TO BE USED*

ENTER METHOD NAMES (ONE PER LINE) HERE.  
TYPE "#STOP" WHEN FINISHED.

1.000=1/14/20  
2.000=1/7R  
2.000=3/14/20  
3.000=#S  
*FORGOT TO END METHOD CODE WITH  
SIGN, SO, LATER, IT IS DONE RIGHT*

1.000=1/14/20/  
2.000=1/7R  
2.000=3/14/20/  
3.000=#S

DO YOU WISH TO HAVE THE DATA DISPLAYED AT THE TERMINAL? YES

WAIT FOR JOB "S3" TO SEE OUTPUT OR TYPE "CTRL-B"  
TO IGNORE DISPLAY

S3 OUTPUT REQUESTED IS:

*THERE IS A SUBSTANTIAL  
WAIT AT THIS POINT (5-30 MINUTES,  
DEPENDING ON THE LOAD ON THE SYSTEM)  
UNILE THE DATA BASE IS READIED FOR  
SEARCHING.*

RECORD = 22  
PROBLEM = 9 (A=100.0,B=0.5 )  
PROBLEM INFO = 202002100200222 000.21 040.60 000.00 010.3

METHOD = 1/14/20/

DATE	TIME	NX	NY	HMAX	NRUNK	ERRMAX	ERRMXF	ERRL2
09/26/78	02.20	5	5	2.50E-01	9	8.3E+00	8.3E+00	4.3E-01
09/26/78	02.20	9	9	1.25E-01	49	9.1E-01	9.5E-01	1.2E-02
09/26/78	02.20	13	13	8.33E-02	121	2.2E-01	2.5E-01	1.5E-03
09/26/78	02.20	17	17	6.25E-02	225	1.1E-01	1.3E-01	5.5E-04
09/26/78	02.20	21	21	5.00E-02	361	6.8E-02	2.1E-02	2.7E-04
11/17/78	18.11	25	25	4.17E-02	529	4.6E-02	9.6E-03	1.6E-04

RESMAX	RESMXR	RESL2	SOLMAX	NIT	MEM	TOTLT	TIME1	TIME2	TIME3
4.4E+02	2.6E+06	2.8E+01	9.3E+00	0	3453	.04	.02	0	.02
3.6E+02	1.9E+04	6.2E+00	1.9E+00	0	6001	.29	.08	0	.20
2.2E+02	1.2E+03	2.8E+00	1.2E+00	0	11557	.96	.19	0	.77
2.3E+02	2.5E+02	1.6E+00	1.1E+00	0	21273	2.38	.33	0	2.04
1.5E+02	9.0E+01	1.1E+00	1.1E+00	0	36301	4.91	.51	0	4.40
8.6E+01	4.7E+01	6.4E-01	1.0E+00	0	57793	9.09	.75	.01	8.34

METHOD = 3/14/20/

DATE	TIME	NX	NY	HMAX	NRUNK	ERRMAX	ERRMXF	ERRL2
09/26/78	04.07	3	3	5.00E-01	36	3.6E+00	3.5E+00	4.0E-01
09/26/78	04.07	5	5	2.50E-01	100	5.7E-01	8.8E-01	4.8E-02
09/26/78	04.07	7	7	1.67E-01	196	4.4E-01	4.4E-01	1.6E-02

RESMAX	RESMXR	RESL2	SOLMAX	NIT	MEM	TOTLT	TIME1	TIME2	TIME3
1.5E+01	8.9E+02	6.9E+00	2.6E+00	0	6235	.46	.08	0	.38
5.4E+01	2.3E+08	6.6E+00	1.6E+00	0	14911	1.99	.24	0	1.76
6.4E+01	3.2E+10	3.5E+00	1.4E+00	0	31331	5.59	.47	0	5.12

ANOTHER SET OF DATA FOLLOWS - DELETED HERE

DO YOU WISH TO HAVE THIS OUTPUT PRINTED AND MAILED? YES

DO YOU WISH TO HAVE THE DATA ANALYZED? YES

Y-AXIS KEYWORD?

THE POSSIBLE CHOICES ARE:

- ERR-NODES
- ERR-GRID
- ERRL2
- RESMAX
- RESL2
- RESMXR
- ? ERR-GRID

X-AXIS KEYWORD?

THE POSSIBLE CHOICES ARE:

- NUMBER
- MEMORY
- TLTIME
- TIME1
- TIME2
- TIME3
- ? TLTIME

A PLOT OF ERR-GRID  
VS  
TOTAL TIME  
IS REQUESTED

THERE IS A WAIT  
HERE WHILE THIS  
JOB RUNS

DO YOU WISH TO HAVE THE DATA PLOTTED? YES

WAIT FOR JOB "S4" TO SEE OUTPUT OR TYPE "CTRL-B"  
TO IGNORE DISPLAY

SA ANALYSIS OUTPUT REQUESTED 103

PILOT  
X=TLTIME  
Y=ERR-BRTO  
RECORDS= 22 23  
METHODS=  
1/14/20/  
3/14/20/

SLOPF ERR-TLTIME  
1/14/20/  
3/14/20/  
5 0/0 TLTIME  
1/14/20/  
3/14/20/  
0.5 0/0 TLTIME  
1/14/20/  
3/14/20/  
0.05 0/0 TLTIME  
1/14/20/  
3/14/20/

AVE RANK	MINIMUM	1ST QUART	MEDIAN	3RD QUART	MAXIMUM
1.50	-8.4E-01	-8.4E-01	-8.4E-01	-8.4E-01	-5.1E-01
1.50	-1.4E+00	-1.4E+00	-1.4E+00	-1.4E+00	-5.9E-01
AVE RANK	MINIMUM	1ST QUART	MEDIAN	3RD QUART	MAXIMUM
1.50	2.6E+00	2.6E+00	2.6E+00	2.6E+00	4.9E+01
1.50	3.7E+01	3.7E+01	3.7E+01	3.7E+01	1.2E+02
AVE RANK	MINIMUM	1ST QUART	MEDIAN	3RD QUART	MAXIMUM
1.50	4.1E+01	4.1E+01	4.1E+01	4.1E+01	4.5E+03
1.50	1.9E+02	1.9E+02	1.9E+02	1.9E+02	6.2E+03
AVE RANK	MINIMUM	1ST QUART	MEDIAN	3RD QUART	MAXIMUM
1.50	6.3E+02	6.3E+02	6.3E+02	6.3E+02	4.2E+05
1.50	1.0E+03	1.0E+03	1.0E+03	1.0E+03	3.1E+05

*These are statistics about the data  
used-GRID for the 2 problems and 2 methods*

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

DO YOU WISH TO HAVE THIS OUTPUT MAILED? Y

FIMM5 = FIM2031 AT MATH

JOB "S3" MAILED

FIMS4 = FIM2031 AT MATH

FIMM6 = FIM2031 AT MATH

JOB "S4" MAILED

*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. (theses 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, you 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: ON LINE 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.

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. Following that is the table that associates record numbers with the problems in the PDE Population [7]

## PDE POPULATION - RECORD KEY

Record Number	Problem Number	MACRO Number	PARAMETERS
1	1		
2	2		
3	3		
4	13		
5	4	1	$\alpha=1.5$
6			$\alpha=2.5$
7			$\alpha=3.5$
8			$\alpha=4.5$
9	5	2	$\alpha= 0$
10			$\alpha= 10$
11			$\alpha= 20$
12			$\alpha= 64$
13			$\alpha=100$
14	6		
15	7	3	$\alpha= 10$
16			$\alpha= 20$
17			$\alpha= 50$
18			$\alpha=100$
19	8	21	$\alpha=0.15$
20	9	4	$\alpha= 10 \quad \beta=0.5$
21			$\alpha= 50 \quad \beta=0.5$
22			$\alpha= 100 \quad \beta=0.5$
23			$\alpha= 500 \quad \beta=0.5$
24			$\alpha=1000 \quad \beta=0.5$
25			$\alpha= 10 \quad \beta=0.117$
26	53	18	$\alpha=30 \quad \beta=20$
27	9	4	$\alpha= 50 \quad \beta=0.117$
28			$\alpha= 100 \quad \beta=0.117$
29			$\alpha= 500 \quad \beta=0.117$
30			$\alpha=1000 \quad \beta=0.117$
31	10		
32	11	5	$\alpha= \pi$
33			$\alpha= 2\pi$

## PDE POPULATION - RECORD KEY

Record Number	Problem Number	MACRO Number	PARAMETERS
34 35 36	11	5	$\alpha = 3\pi$ $\alpha = 5\pi$ $\alpha = 10\pi$
37	39	13	$\alpha = 1$ $\beta = 8$ $\gamma = -1$
38 39 40 41 42 43	12	6	$\alpha = \pi$ $\beta = \pi$ $\alpha = \pi$ $\beta = 10$ $\alpha = 10$ $\beta = \pi$ $\alpha = 20$ $\beta = \pi$ $\alpha = 10$ $\beta = 10$ $\alpha = 10$ $\beta = 20$
44	36	22	$\alpha = 0$ $\beta = 0$
45	37		
46 47	19	7	$\alpha = 0.15$ $\alpha = 0.25$
48 49 50 51	30	8	$\alpha = 1.5$ $\alpha = 2.5$ $\alpha = 3.5$ $\alpha = 4.5$
52	22		
53	32		
54 55 56	53	18	$\alpha = 10$ $\beta = \pi$ $\alpha = 10$ $\beta = 10$ $\alpha = 30$ $\beta = 10$
57	21		
58 59	5	2	$\alpha = 5$ $\alpha = 8$
60 61 62	17	10	$\alpha = 1$ $\beta = 2$ $\alpha = 5$ $\beta = 3$ $\alpha = 8$ $\beta = 5$
63 64	20	11	$\alpha = 0$ $\beta = \text{HOMOGENEOUS}$ $\alpha = 10$ $\beta = \cancel{\gamma}$
65	31		

PDE POPULATION - RECORD KEY

Record Number	Problem Number	MACRO Number	PARAMETERS
66	33		
67	34		
68	35		
69 70 71	38	12	$\alpha=3$ $\alpha=5$ $\alpha=7$
72 73 74 75 76 77 78 79	39	13	$\alpha= 1 \quad \beta= 2 \quad \gamma= 5$ $\alpha= 23 \quad \beta= 2 \quad \gamma= 1$ $\alpha= 10 \quad \beta=11 \quad \gamma= 0$ $\alpha=100 \quad \beta= 2 \quad \gamma= 2$ $\alpha= 10 \quad \beta= 4 \quad \gamma= 1$ $\alpha= 4 \quad \beta= 5 \quad \gamma=-0.5$ $\alpha= 3 \quad \beta= 6 \quad \gamma= 2$ $\alpha=0.5 \quad \beta= 3 \quad \gamma=10$
80 81 82 83	15	14	$\alpha=0.2 \quad \beta=1.4 \quad \epsilon=0.10$ $\alpha=1.0 \quad \beta=2.5 \quad \epsilon=0.10$ $\alpha=0.2 \quad \beta=1.5 \quad \epsilon=0.04$ $\alpha=0.2 \quad \beta=2.5 \quad \epsilon=0.04$
84 85 86	16	15	$\alpha=0.0$ $\alpha=0.1$ $\alpha=1.0$
87 88 89	18	16	$\alpha=1 \quad \beta=2$ $\alpha=5 \quad \beta=3$ $\alpha=8 \quad \beta=5$
90 91 92 93 94 95	23	17	$k=1 \quad a=387.75 \quad b=50.0$ $k=1 \quad a=554.50 \quad b= 0.554$ $k=2 \quad a=387.75 \quad b=50.0$ $k=2 \quad a=554.50 \quad b= 0.554$ $k=3 \quad a=387.75 \quad b=50.0$ $k=3 \quad a=554.50 \quad b= 0.554$
96	43		
97	44		
98 99 100	1	19	$\alpha= 0.1$ $\alpha= 1.0$ $\alpha=10.0$

## PDE POPULATION - RECORD KEY

Record Number	Problem Number	MACRO Number	PARAMETERS
101 102 103	3	20	$\alpha=0.1$ $\alpha=1.0$ $\alpha=10.0$
104 105 106	8	21	$\alpha=0.10$ $\alpha=0.35$ $\alpha=0.45$
107 108 109	36	22	$\alpha=0.1 \quad \beta=-0.5$ $\alpha=0.0 \quad \beta=1.0$ $\alpha=0.5 \quad \beta=0.0$
110 111 112	35	23	$\alpha=0.1$ $\alpha=1.0$ $\alpha=10.0$
113 114 115	40	24	$\alpha=1$ $\alpha=3$ $\alpha=7$
116 117 118	42	25	$\alpha=0 \quad \beta=1 \quad \gamma=1$ $\alpha=-1 \quad \beta=2 \quad \gamma=2$ $\alpha=-1 \quad \beta=2 \quad \gamma=5$
119 120 121	41	26	$\alpha=-10 \quad \beta=5$ $\alpha=1 \quad \beta=10$ $\alpha=-10 \quad \beta=25$
122 123 124 125 126	45	28	A=1 $\epsilon^2=0.50 \quad g(x)=1/x$ A=2 $\epsilon^2=1.00 \quad g(x)=1/x$ A=3 $\epsilon^2=0.25 \quad g(x)=\text{Exp}(x)$ A=4 $\epsilon^2=0.50 \quad g(x)=\text{Exp}(x)$ A=5 $\epsilon^2=1.00 \quad g(x)=\text{Exp}(x)$
127 128 129 130 131	48	29	A=1 $p=1 \quad \beta=.04 \quad \gamma=2 \quad \phi=1.425$ A=2 $p=1 \quad \beta=.04 \quad \gamma=25 \quad \phi=1.425$ A=3 $p=2 \quad \beta=.04 \quad \gamma=2 \quad \phi=1.425$ A=4 $p=2 \quad \beta=.04 \quad \gamma=25 \quad \phi=1.425$ A=5 $p=2 \quad \beta=.50 \quad \gamma=2 \quad \phi=1.425$
132 133 134 135	49	30	A=1 $p=1 \quad \beta=.50 \quad \gamma=2 \quad \phi=1.425$ A=2 $p=1 \quad \beta=.50 \quad \gamma=25 \quad \phi=1.425$ A=3 $p=2 \quad \beta=.04 \quad \gamma=2 \quad \phi=1.425$ A=4 $p=2 \quad \beta=.50 \quad \gamma=2 \quad \phi=1.425$
136	50	31	A=1 $p=1 \quad \beta=.50 \quad \gamma=2 \quad \phi=1.425$

## PDE POPULATION - RECORD KEY

Record Number	Problem Number	MACRO Number	PARAMETERS
137	50	31	A=2 p=1 $\beta=.50$ $\gamma= 2$ $\phi=10.0$
138			A=3 p=2 $\beta=.04$ $\gamma=25$ $\phi= 1.425$
139			A=4 p=2 $\beta=.50$ $\gamma= 2$ $\phi= 1.425$
140	51	32	A=1 p=1 $\phi^2= 2$
141			A=2 p=1 $\phi^2=1000$
142			A=3 p=2 $\phi^2= 2$
143	27		
144	52	33	A=1 $\theta=10$ $\phi^2= 2$
145			A=2 $\theta=10$ $\phi^2= 4$
146			A=3 $\theta=10$ $\phi^2=49$
147	28	34	A=1 $\alpha= 1$
148			A=2 $\alpha= 10$
149			A=3 $\alpha=100$
150	24	35	A=1 $\beta= 1$
151			A=2 $\beta=10$
152	25	36	A=1 $k=1/4$ $\epsilon= 100$ $c=-1/10$
153			A=2 $k=1/4$ $\epsilon=1000$ $c=-1/10$
154			A=3 $k=1/2$ $\epsilon= 1$ $c=-1/100$
155			A=4 $k=1/2$ $\epsilon= 10$ $c=-1/10$
156			A=5 $k=1$ $\epsilon= 1$ $c=-1/10$
157			A=6 $k=1$ $\epsilon= 10$ $c=-1/10$
158			A=7 $k=1$ $\epsilon= 100$ $c=-1$
159			A=8 $k=1$ $\epsilon=1000$ $c=-1$
160	26	37	A=1 $L= 1$
161			A=2 $L= 5$
162			A=3 $L=10$
163	29	38	A=1 $K=-3$
164			A=2 $K=-1$
165			A=3 $K= 1$
166			A=4 $K= 3$
167			A=5 $K= 5$
168	46	39	A=1 $\alpha=1$ $\beta= 2$
169			A=2 $\alpha=4$ $\beta= 2$
170			A=3 $\alpha=4$ $\beta=10$
171			A=4 $\alpha=8$ $\beta= 2$
172	47	40	A=1 $N=1$

## PDE POPULATION - RECORD KEY

Record Number	Problem Number	MACRO Number	PARAMETERS
173 174	47	40	A=2 N=4 A=3 N=8
175 176 177 178	56	41	A=1 L=0.3 A=2 L=0.5 A=3 L=0.6 A=4 L=0.7
179 180 181 182	57	42	A=1 KASE=1 BREAK=1 DECAY1=3 A=2 KASE=1 BREAK=3 DECAY1=2 A=3 KASE=2 BREAK=1 DECAY1=3 A=4 KASE=2 BREAK=6 DECAY1=2
183 184	54	27	A=1 $\alpha=0.5$ A=2 $\alpha=0.9$

All of the modules currently available in ELLPACK 77 are described briefly in this appendix. The 31 modules are listed below.

<u>Discretization</u>	<u>Indexing</u>	<u>Solution</u>
1. 5-POINT STAR	14. NATURAL	18. BAND SOLVE
2. 7-POINT 3D	15. RED-BLACK	19. SYMMETRIC BAND
3. PC-C1 COLLOCATION	16. YALE MIN DEG	20. LINPACK BAND
4. P3-C1 GALERKIN	17. YALE RCM	21. LINPACK SPD BAND
5. HODIE-HELMHOLTZ		22. SPARSE GE-PIVOTING
6. HODIE-ACF		23. SOR
7. HODIE-ACDEF		24. JACOBI SI
8. HODIE 27-POINT 3D		25. JACOBI CG
9. FFT 9-POINT		26. SYMMETRIC SOR SI
10. 2DEPEP		27. SYMMETRIC SOR CG
11. MARCHING ALGORITHM		28. REDUCED SYSTEM SI
12. DYAKANOV-CG		29. REDUCED SYSTEM CG
13. DYAKANOV-CG 4		30. YALE SPARSE
		31. YALE ENVELOPE

## REFERENCES

- [1] R. F. Boisvert, XEQ-ELLPACK, CSDTR XXX, Computer Science Department, Purdue University, August, 1979
- [2] Purdue University Computing Center, Introduction to the Purdue Editor, LO-LEARNPED, June, 1979.
- [3] Purdue University Computing Center, Beginners Guide to the Purdue Operating System PIRATE, LO-PIRATE, August, 1978.
- [4] Purdue University Computing Center, A description of control cards and job processing, LO-CONTROL, March, 1979
- [5] G. Sewell, ELLPACK data analysis program, 2 pages, May, 1979.
- [6] J. R. Rice, ELLPACK 77 Users Guide, CSD-TR 289, Computer Science Department, Purdue University, September, 1978
- [7] E. N. Houstis and J. R. Rice, A Population of PDEs for elliptic, linear problems on rectangles, to appear (Preliminary and partial version: CSD-TR 263, May 1978)