Research Note 81-17

AN ANALYTICAL MODEL FOR DEVELOPING OBJECTIVE MEASURES OF AIR CREW PROFICIENCY WITH MULTIVARIATE TIME SEQUENCED DATA

VOLUME II. COMPUTER PROGRAM DOCUMENTATION

Edward M. Connelly, Pamela Johnson
Performance Measurement Associates, Inc.

Brian D. Shipley, Jr.
Army Research Institute

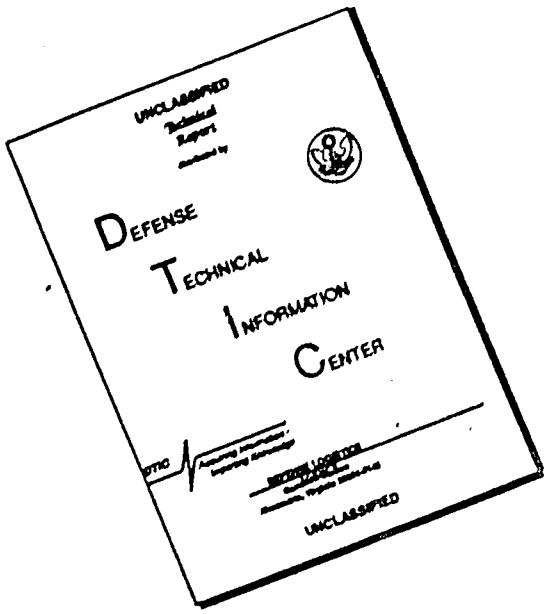
ARI FIELD UNIT AT FORT RUCKER, ALABAMA

U. S. Army

Research Institute for the Behavioral and Social Sciences

May 1981

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: Research Note 81-17	2. GOVT ACCESSION NO.: AD A127 313
	3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): An Analytical Model for Developing Objective Measures of Air Crew Proficiency with Multivariate Time Sequenced Data. Volume II. Computer Program Documentation	5. TYPE OF REPORT & PERIOD COVERED: Final - Jan 1980- May 81
	6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): Edward M. Connelly, Pamela Johnson (PMA), and Brian D. Shipley (ARI)	8. CONTRACT OR GRANT NUMBER(s): MDA 903-80-C-0198
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Performance Measurement Associates, Inc. 410 Pine Street, S.E. Vienna, Virginia 22333	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 2Q162717A790
11. CONTROLLING OFFICE NAME AND ADDRESS: Army Research Institute for the Behavioral and Social Sciences. 5001 Eisenhower Avenue, Alexandria, Virginia 22333	12. REPORT DATE: May 1981
	13. NUMBER OF PAGES:
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Nap-of-the-earth flight
Night flight
Night vision
Night reconnaissance

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This volume contains computer programs used to analyze the subject data, procedures for the use of the programs, and example outputs. These programs are the result of a theoretical investigation of analytic methods for deriving differential weighting functions from preselected samples of multivariate, time sequenced observations of aircrew performance. The research effort resulted in an analytic model which could be used to prepare and to further investigate differential weighting functions as a means of establishing relationships between time sequenced observations of aircrew performance (over)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

item 20. continued

and independent assessments of aircrew proficiency, (Volume I. Analysis and Results, RN 81-16).

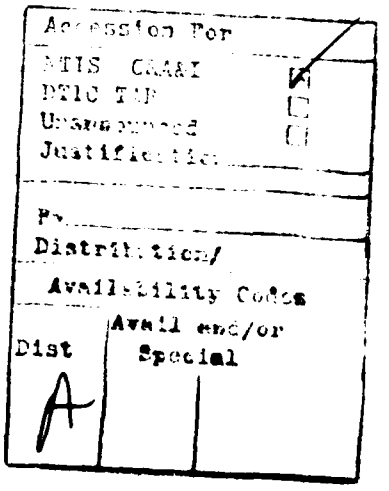


TABLE OF CONTENTS

TABLE OF CONTENTS (CONTINUED)

LIST OF TABLES

INTRODUCTION

Volume II, Computer Program Documentation, was prepared under Contract No. MDA 903-80-C-0198, entitled "An Analytical Model for Developing Objective Measures of Air Crew Proficiency with Multivariate Time Sequenced Data." This second volume provides documentation of the basic computer programs used in the processor termed "Measurement Analysis Processor" (MAP).

The computer programs were designed to make extensive use of subprograms so that common functions used in the various programs could also use common subroutines. Furthermore, the structured approach to programming was used - to the degree that the unstructured FORTRAN language would permit. However, in the initial programming efforts limited use of the "go to" statements was maintained; but it was necessary for program variation to insert some "go to" statements which of course complicated the program structure. We hope this additional complication will not lead to difficulties in using or modifying the programs.

This volume contains programs used to analyze the subject data, procedures for use of the programs, and example outputs.

On the procedure page, the user inputs are examples from data files at PMA.

The computer used is an LSI/11, utilizing floppy discs containing programs and data. The DK: referred to on the procedure page corresponds to the disc drive used by the machine operator.

The subscripts indicating footnotes on the procedure page do not appear on the screen as computer outputs.

Program Listing INTHEX

```
C PROGRAM 'INTHEX'
C PURPOSE: CONVERT C-50 DATA FILE TO HEXIDECIMAL NOTATION
C AND FORMAT DATA FOR TRANSMISSION OVER PHONE LINES
C USER SPECIFIES FILE NAME, BEGINNING LOCATION, AND
C NUMBER OF BLOCKS TO PROCESS
      INCLUDE SYSCOMXKEYS.F
      DIMENSION INBUF(15240),IOUTB(508),NAME1(8),
     1          IHTAB(257)
      INTEGER*4 IF1POS
      DATA IHTAB/'00','01','02','03','04','05','06','07',
     1           '08','09','0A','0B','0C','0D','0E','0F',
     1           '10','11','12','13','14','15','16','17',
     1           '18','19','1A','1B','1C','1D','1E','1F',
     1           '20','21','22','23','24','25','26','27',
     1           '28','29','2A','2B','2C','2D','2E','2F',
     1           '30','31','32','33','34','35','36','37',
     1           '38','39','3A','3B','3C','3D','3E','3F',
     1           '40','41','42','43','44','45','46','47',
     1           '48','49','4A','4B','4C','4D','4E','4F',
     1           '50','51','52','53','54','55','56','57',
     1           '58','59','5A','5B','5C','5D','5E','5F',
     1           '60','61','62','63','64','65','66','67',
     1           '68','69','6A','6B','6C','6D','6E','6F',
     1           '70','71','72','73','74','75','76','77',
     1           '78','79','7A','7B','7C','7D','7E','7F',
     1           '80','81','82','83','84','85','86','87',
     1           '88','89','8A','8B','8C','8D','8E','8F',
     1           '90','91','92','93','94','95','96','97',
     1           '98','99','9A','9B','9C','9D','9E','9F',
     1           'A0','A1','A2','A3','A4','A5','A6','A7',
     1           'A8','A9','AA','AB','AC','AD','AE','AF',
     1           'B0','B1','B2','B3','B4','B5','B6','B7',
     1           'B8','B9','BA','BB','BC','BD','BE','BF',
     1           'C0','C1','C2','C3','C4','C5','C6','C7',
     1           'C8','C9','CA','CB','CC','CD','CE','CF',
     1           'D0','D1','D2','D3','D4','D5','D6','D7',
     1           'D8','D9','DA','DB','DC','DD','DE','DF',
     1           'E0','E1','E2','E3','E4','E5','E6','E7',
     1           'E8','E9','EA','EB','EC','ED','EE','EF',
     1           'F0','F1','F2','F3','F4','F5','F6','F7',
     1           'F8','F9','FA','FB','FC','FD','FE','FF','0F'/
C NAME INPUT FILE
      WRITE(1,1010)
      READ(1,1011) (NAME1(I),I=1,8)
C ENTER STARTING WORD NUMBER
      WRITE(1,1030)
      READ(1,1031) IF1POS
C ENTER NUMBER OF BLOCKS (#BLOCKS < = 29) TO READ
      WRITE(1,1040)
      READ(1,1041) NUMBLK
      NUMWRD=NUMBLK*254
C OPEN INPUT FILE
      CALL SPCHOS(KSOREAD,NAME1,16,1,ITYPE1,ICODE1)
      IF(ICODE1.NE.0) GO TO 300
C POSITION INPUT FILE TO BEGINNING WORD LOCATION
```

```
C AND READ IN WORDS [illegible]
      CALL PRNFSS(KSREAD,#FREA.1,LOC(INBUF),NUMWRD,IFIPOS,
     1           NAME,IPC(1))
      IF(IPC(1).NE.0) GO TO 500
C CLOSE INPUT FILE
      CALL SRCHSS(KSCLOS,0,0,1,0,ICODE1)
      IF(ICODE1.NE.0) GO TO 600
C PROCESS INPUT WORDS AND WRITE LISTING
      WRITE(1,1050)
      READ(1,1051) IA1
      IF(IA1.NE.'GO') GO TO 900
      K=0
      DO 200 J=1,NUMBLK
      DO 100 J=1,254
      K=K+1
      INTVAR=INBUF(K)
      CALL INTHEX(INTVAR,DIGIT,IHEX1,IHEX2)
      L1=(J-1)*2+1
      L2=L1+1
      IOUTB(L1)=IHEX1
      IOUTB(L2)=IHEX2
  100 CONTINUE
      WRITE(1,1200) (IOUTB(JJ),JJ=9,508)
  200 CONTINUE
      GO TO 900
  300 CONTINUE
      WRITE(1,1110)
      GO TO 900
  500 CONTINUE
      WRITE(1,1130)
      GO TO 900
  600 CONTINUE
      WRITE(1,1140)
      GO TO 900
  900 CONTINUE
      CALL EXIT
 1010 FORMAT(' NAME INPUT FILE')
 1011 FORMAT(A2)
 1030 FORMAT(' ENTER BEGINNING WORD (START=WORD(0))')
 1031 FORMAT(I8)
 1040 FORMAT(' ENTER NUMBER OF BLOCKS TO READ (1 < = # BLOCKS < = 29)')
 1041 FORMAT(I8)
 1050 FORMAT(' ENTER "GO" TO START TRANSMISSION AFTER'/
     1       ' HITTING FUNCTION KEY #4 AND TRANSMISSION'/
     1       ' COMMAND IS REQUESTED')
 1051 FORMAT(A2)
 1110 FORMAT(' OPEN FAILED FOR INPUT FILE')
 1130 FORMAT(' POSITION READ FAILED FOR INPUT FILE')
 1140 FORMAT(' CLOSE FAILED FOR INPUT FILE')
 1200 FORMAT(400A2)
      END
      SUBROUTINE INTHEX(INTVAR,IHEX1,IHEX2)
      DIMENSION IHTHEX(16)
      IF(INTVAR.EQ.[illegible]) GO TO 500
      NBITS(16)=0
      IF(INTVAR.LT.[illegible])
```

Program Listing INTHEX (Concluded)

```
      IHOLD=IABS(INTVAL
      DO 200 I=1,15
      I2=16-I
      IPOW2=2**(I2-1)
      ICHK=IHOLD-IPOW2
      IF(ICHK.LT.0) GO TO 100
      IBITS(I2)=1
      IHOLD=ICHK
      GO TO 200
  100 CONTINUE
      IBITS(I2)=0
  200 CONTINUE
      DO 400 I=1,2
      JSTART=(I-1)*8+1
      JSTOP=JSTART+7
      IADDR=0
      DO 300 J=JSTART,JSTOP
      J2=J-1-(I-1)*8
      IF(IBITS(J).EQ.1) IADDR=IADDR+2**J2
  300 CONTINUE
      IADDR=IADDR+1
      IF(I.EQ.1) IHEX2=IHTAB(IADDR)
      IF(I.EQ.2) IHEX1=IHTAB(IADDR)
  400 CONTINUE
      GO TO 600
  500 CONTINUE
      IHEX1=IHTAB(257)
      IHEX2=IHTAB(256)
  600 CONTINUE
      RETURN
      END
```

Subroutine INTHEX

```
FORTRAN IV     HB1A-1    FRI 01-MAY-81 01:43:36          PAGE 001

0001          SUBROUTINE INTHEX(IHTAB,INTVAR,IHEX1,IHEX2)
0002          DIMENSION IHTAB(257),IBITS(16)
0003          IF(INTVAR.EQ.-32768) GO TO 500
0005          IBITS(16)=0
0006          IF(INTVAR.LT.0) IBITS(16)=1
0008          IHOLD=IABS(INTVAR)
0009          DO 200 I=1,15
0010          I2=16-I
0011          IPOW2=2**(I2-1)
0012          ICHK=IHOLD-IPOW2
0013          IF(ICHK.LT.0) GO TO 100
0015          IBITS(I2)=1
0016          IHOLD=ICHK
0017          GO TO 200
0018      100 CONTINUE
0019          IBITS(I2)=0
0020      200 CONTINUE
0021          DO 400 I=1,2
0022          JSTART=(I-1)*8+1
0023          JSTOP=JSTART+7
0024          IADDR=0
0025          DO 300 J=JSTART,JSTOP
0026          J2=J-1-(I-1)*8
0027          IF(IBITS(J).EQ.1) IADDR=IADDR+2**J2
0029      300 CONTINUE
0030          IADDR=IADDR+1
0031          IF(I.EQ.1) IHEX2=IHTAB(IADDR)
0033          IF(I.EQ.2) IHEX1=IHTAB(IADDR)
0035      400 CONTINUE
0036          GO TO 600
0037      500 CONTINUE
0038          IHEX1=IHTAB(25
0039          IHEX2=IHTAB(25
0040      600 CONTINUE
0041          RETURN
0042          END
```

Table 1

INTHEX

Integer to Hexidecimal	
Purpose: To convert a data file to hexidecimal notation for better transmission from time-sharing facilities. Note: The procedure will change with different time-sharing companies.	
Computer Output	User Input

Program Listing HEXINT

```
FORTRAN IV     H01A-1    FRI 01-NOV-01 00:54:07              PAGE 001

      C PROGRAM 'HEXINT.FOR'
      C PURPOSE: CONVERT HEXIDECIMAL DATA FILE READ FROM DIALCOM
      C BACK TO BINARY UNFORMATTED (PROGRAM INTHEX AT DIALCOM)
      C CONTRACT C-50
      C ARMY RESEARCH INSTITUTE
      C 06-AUG-80
0001        DIMENSION IDATA1(250,4),IDATA2(250),IHTAB(16,5),IHEX(4)
0002        LOGICAL*1 IDATA1,IHTAB,IHEX,IGG
0003        DATA IHTAB/'0','1','2','3','4','5','6','7',
           1          '8','9','A','B','C','D','E','F',
           1          0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,
           1          0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1,
           1          0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,
           1          0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1/,
           1     IGG/'G'/
0004        PAUSE 'INSERT I-O DATA DISC IN UNIT ONE (DK:) AND HIT RETURN'
0005        WRITE(7,7010)
0006        CALL ASSIGN(2,'  ',-1,'RDO',',NC',1)
0007        WRITE(7,7020)
0008        CALL ASSIGN(3,'  ',-1,'NEW',',NC',1)
0009        RBIGN=(-32768.0)
0010        IBIGN=INT(RBIGN)
0011        IOUT=0
0012        IN=0
0013        IDUM1=1234
0014        IDUM2=4321
0015        IDUM3=5678
0016        IDUM4=0
0017        IDUM5=5634
0018        IDUM6=0
0019      200 CONTINUE
0020        READ(2,100,END=500) ((IDATA1(I,J),J=1,4),I=1,250)
0021        IN=IN+1
0022        DO 210 I=1,250
0023        IHEX(1)=IDATA1(I,1)
0024        IF(IHEX(1).EQ.IGG) GO TO 205
0026        IHEX(2)=IDATA1(I,2)
0027        IHEX(3)=IDATA1(I,3)
0028        IHEX(4)=IDATA1(I,4)
0029        CALL HEXIN(IHEX,INTVAR)
0030        IDATA2(I)=INTVAR
0031        GO TO 210
0032      205 CONTINUE
0033        IDATA2(I)=IBIGN
0034      210 CONTINUE
0035        WRITE(3) IDUM1,IDUM2,IDUM3,IDUM4,IDUM5,IDUM6,(IDATA2(I),I=1,250)
0036        IOUT=IOUT+1
0037        WRITE(7,*) IOUT
0038        WRITE(7,*)    (IDATA2(I),I=1,250)
0039        GO TO 200
0040      500 CONTINUE
0041        CALL CLOSE(2)
0042        CALL CLOSE(3)
0043        WRITE(7,*)    IOUT
```

Program Listing HEXINT (Concluded)

```
FORTRAN IV     H01A-1    FRI 01-MAY-81 00:54:07              PAGE 002

0044          STOP 'END OF PROGRAM HEXINT.FOR'
0045    2010 FORMAT(80A1)
0046    7010 FORMAT(' NAME INPUT FILE (ASSIGN TO DK:)'/)
0047    7020 FORMAT(' NAME OUTPUT FILE (ASSIGN TO DK:)'/)
0048    7030 FORMAT(' INPUT RECORDS: ',I7/
     1       ' OUTPUT RECORDS: ',I7)
0049    7040 FORMAT(' BLOCK: ',I4)
0050    7050 FORMAT(1X,1017)
0051          END
```

```
FORTRAN IV     H01A-1    FRI 01-MAY-81 00:54:21          PAGE 001

0001          SUBROUTINE HEXINT(IHTAB,IHEX,INTVAL)
0002          DIMENSION IHTAB(16,5),IHEX(4),IBITS(16)
0003          LOGICAL*1 IHTAB,IHEX
0004          L=17
0005          DO 400 I=1,4
0006          DO 100 J=1,16
0007          IF(IHEX(I).EQ.IHTAB(J,1)) GO TO 200
0009      100 CONTINUE
0010          J=16
0011      200 CONTINUE
0012          JSAVE=J
0013          DO 300 K=2,5
0014          L=L-1
0015          IBITS(L)=IHTAB(JSAVE,K)
0016      300 CONTINUE
0017      400 CONTINUE
0018          INTVAL=0
0019          DO 500 M=1,15
0020          M1=M-1
0021          IF(IBITS(M).EQ.1) INTVAL=INTVAL+2**M1
0023      500 CONTINUE
0024          IF(IBITS(16).EQ.1) INTVAL=(-INTVAL)
0026          RETURN
0027          END
```

Table 2

HEXINT.FOR

Hexidecimal to Integer	
Purpose: To convert a hexidecimal data file to an integer data file.	
Computer Output	User Input
	• Run HEXINT 〈CR〉
Pause--I/O data disc in unit one (DK:) and hit return.	〈CR〉
Name input file (assign to DK:)	*DK:TTEST3.DAT 〈CR〉
Name output file (assign to DK:)	*DK:TTEST4.DAT 〈CR〉

Program Listing FMERGE

```
FORTRAN IV     V01A-1    FRI 01-MAY-81 08:59:45          PAGE 001

      C PROGRAM 'FMERGE.FOR'
      C PURPOSE: MERGE C-50 DATA FILES BY STUDENT NUMBER
      C CONTRACT C-50
      C NAVY RESEARCH INSTITUTE
      C FORT RUCKER FIELD UNIT
      C 26-JULY-80
0001        DIMENSION IDATA(250)
0002        PAUSE 'INSERT OUTPUT DATA DISC IN UNIT ONE (DK:) AND HIT RETURN'
0003        WRITE(7,7020)
0004        CALL ASSIGN(3,' ',-1,'NEW','NC',1)
0005        WRITE(7,7060)
0006        READ(5,5060) ISBJCT
0007        IOUT=0
0008        IN=0
0009    100 CONTINUE
0010        WRITE(7,7011)
0011        READ(5,5020) IFINSH
0012        IF(IFINSH.EQ.1) GO TO 900
0014        PAUSE 'INSERT INPUT DATA DISC IN UNIT ZERO (SY:) AND HIT RETURN'
0015        WRITE(7,7010)
0016        CALL ASSIGN(2,' ',-1,'RDO','NC',1)
0017    200 CONTINUE
0018        READ(2,END=300,ERR=300) IDUM1,IDUM2,IDUM3,IDUM4,IDUM5,IDUM6,
           1                        (IDATA(I),I=1,250)
0019        IN=IN+1
0020        WRITE(7,7100) IN,IOUT,IDATA(2),IDATA(227)
0021        IF((IDATA(2).NE.ISBJCT).OR.(IDATA(227).NE.ISBJCT)) GO TO 200
0023        WRITE(3) IDUM1,IDUM2,IDUM3,IDUM4,IDUM5,IDUM6
           1                (IDATA(I),I=1,250)
0024        IOUT=IOUT+1
0025        GO TO 200
0026    300 CONTINUE
0027        CALL CLOSE(2)
0028        GO TO 100
0029    900 CONTINUE
0030        CALL CLOSE(3)
0031        WRITE(7,7004) IN,IOUT
0032        STOP 'END OF PROGRAM: FMERGE.FOR'
0033    5020 FORMAT(I7)
0034    5060 FORMAT(I7)
0035    7010 FORMAT(' NAME INPUT FILE (ASSIGN TO SY:) ')
0036    7011 FORMAT(' ENTER 0 (ZERO) TO READ ANOTHER FILE '
           1    ' OR ENTER 1 (ONE) TO CLOSE OUTPUT FILE ')
0037    7020 FORMAT(' NAME OUTPUT FILE (ASSIGN TO DK:) ')
0038    7030 FORMAT(' INPUT RECORDS: ',I7/
           1    ' OUTPUT RECORDS: ',I7)
0039    7060 FORMAT(' ENTER STUDENT NUMBER (I) ')
0040    7100 FORMAT(1X,I6,1X,I6,1X,I7,1X,I7)
0041        END
```

Table 3

FMERGE.FOR

Merge Data Files by Student Number	
Purpose: The program FMERGE is used when a subject's data is located in two different areas (i.e., 2 different discs). FMERGE will bring both sets of data together to form a single subject file.	
Computer Output	User Input
	• Run FMERGE ⟨CR⟩
Pause - Insert output data disc in unit one (DK:) and hit RETURN	⟨CR⟩
Name Output File	*DK:TTEST3.DAT ⟨CR⟩
Enter Student Number (I)[1]	20935 ⟨CR⟩
Enter 0 (zero) to read another file or enter 1 (one) to close output file	[1] ⟨CR⟩

[1] Integer

Program Listing DFLIST

```
FORTRAN IV     HB1A-1    FRI 01-MAY-81 01:10:42          PAGE 001

      C PROGRAM 'DFLIST.FOR'
      C LIST THE C-50 DATA FILE
0001        DIMENSION X(29,10)
0002        INTEGER X
0003        NBLOCK=0
0004        PAUSE 'INSERT DATA DISC IN DK: AND HIT RETURN'
0005        WRITE(7,7000)
0006        CALL ASSIGN(2,' ',-1,'KOO','NC',1)
0007        WRITE(7,7010)
0008        READ(7,7011) IUNIT
0009    100 CONTINUE
0010        CALL INPUT(X,NBLOCK,MARK,2)
0011        IF(MARK.EQ.-2) GO TO 999
0013        DO 200 I=1,29
0014        WRITE(IUNIT,7020) NBLOCK,I,(X(I,J),J=1,10)
0015    200 CONTINUE
0016        GO TO 100
0017    999 CONTINUE
0018        STOP
0019   7000 FORMAT(' NAME INPUT FILE: ')
0020   7010 FORMAT(' ENTER ',A1,' NUMBER FOR LISTING (6,7): ')
0021   7011 FORMAT(I2)
0022   7020 FORMAT(' ',I3,',',I2,',',10I7)
0023        END
```

```
FORTRAN IV     H01A-1    FRI 01-MAY-81 01:10:54                PAGE 001

0001          SUBROUTINE INPUT(X,NBLOCK,MARK,N)
0002          INTEGER DATA,X
0003          DIMENSION DATA(256),X(25,10)
0004          IP=0
0005          IPL=0
0006          DO 10 J=1,10
0007          DO 10 I=1,25
0008          X(I,J)=0
0009       10 CONTINUE
0010          MARK=0
0011          READ(N,ERR=50,END=60) IDUM1,IDUM2,IDUM3,IDUM4,IDUM5,IDUM6,
              1                     (DATA(I),I=1,250)
0012          NBLOCK=NBLOCK+1
0013          K=0
              C MAIN LOOP
0014          DO 100 J=1,10
0015          K=K+2
              C STUDENT NUMBER
0016          X(1,J)=DATA(K+1)
              C PERIOD
0017          K=K+2
0018          X(2,J)=MOD(DATA(K),256)
              C SEGMENT NUMBER
0019          K=K+10
0020          DO 20 I=1,2
0021          K=K+1
0022          NTK=MOD(DATA(K),2)
0023          X(3,J)=X(3,J)+NTK
0024          DATA(K)=DATA(K)-NTK
0025       20 CONTINUE
0026          K=K-12
0027       30 LPS=DATA(K+12)
0028          IF(IPL.LT.LPS) GO TO 40
0029          IP=IP+256
0030          GO TO 30
0031       40 IPL=LPS
0032          X(4,J)=LPS
              C STICK X (ROLL INPUT  SCALING IS INCHES TIMES 100 MAX=6.35
0033          DO 50 I=5,6
0034          K=K+1
0035          Y=FLOAT(DATA(K))*0.019844+ISIGN(5,DATA(K))/10.0
0036          X(I,J)=INT(Y)
0037       50 CONTINUE
              C STICK Y (PITCH INPUT  SCALING IS INCHES TIMES 100 MAX=6.350
0038          DO 60 I=9,10
0039          K=K+1
0040          Y=FLOAT(DATA(K))*0.019323+ISIGN(5,DATA(K))/10.0
0041          X(I,J)=INT(Y)
0042       60 CONTINUE
0043          DO 70 I=7,8
0044          K=K+1
              C SIDE TASK STATUS
0045          NTK=MOD(DATA(K),2)
0046          DATA(K)=DATA(K)-NTK
```

Subroutine INPUT (Continued)

```
FORTRAN IV     H01A-1    FRI 01-MAY-81 01:10:54              PAGE 002

0848          X(26,J)=X(26,J)+100*NTK
      C PEDAL POSTION SCALING IS INCHES TIMES 100 MAX=3.25"
0849          Y=FLOAT(DATA(K))+0.009919*ISIGN(5,DATA(K))/10.0
0850          X(1,J)=INT(Y)
0851        70 CONTINUE
      C PITCH INDICATED MAX=60 DEGREES, SCALED TIMES 100
0852          DO 80 I=15,16
0853          K=K+1
0854          DATA(K)=DATA(K)+X00(DATA(K),2)
0855          Y=FLOAT(DATA(K))+0.54935*ISIGN(5,DATA(K))/10.0
0856          X(1,J)=INT(Y)
0857        80 CONTINUE
      C ROLL INDICATED MAX=90 DEGREES, SCALED TIMES 100
0858          DO 90 I=17,16
0859          K=K+1
0860          DATA(K)=DATA(K)+X00(DATA(K),2)
0861          Y=FLOAT(DATA(K))+0.54932*ISIGN(5,DATA(K))/10.0
0862          X(1,J)=INT(Y)
0863        90 CONTINUE
      C ALTITUDE
0864          K=K+1
0865          X(19,J)=DATA(K)
      C TORQUE   0 TO 100 PSI
0866          K=K+1
0867          X(20,J)=DATA(K)*5E
      C AIRSPEED MAX=146.7 KNOTS SCALED TIMES 100
0868          X(21,J)=6*M00(DATA(K),256)
      C HEADING 0 TO 360 DEGREES SCALED TIMES 100
0869          K=K+1
0870          X(22,J)=FLOAT(DATA(K))+0.54932*ISIGN(5,DATA(K))/10.0
      C VERTICAL VELOCITY
0871          K=K+1
0872          X(23,J)=DATA(K)
      C YAW 0 TO 14.9 DEGREES/SEC SCALED TIMES 100
0873          K=K+1
0874          X(24,J)=FLOAT(DATA(K))+0.05454*ISIGN(5,DATA(K))/100.0
      C COURSE DEVIATION
0875          K=K+1
0876          X(25,J)=FLOAT(DATA(K))+0.54*ISIGN(5,DATA(K))/10.0
      C SIDE TASK NUMBER
0877          K=K-21
0878          X(26,J)=M00(DATA(K),256)
      C GENERATOR RECTIFIER
0879          X(29,J)=DATA(K)*5E
      C STATION SELECT
0880          K=K+22
0881          X(27,J)=DATA(K)*5E
      C MIKE RESPONSE
0882          X(28,J)=M00(DATA(K),256)
0883        100 CONTINUE
0884          MARK=10
0885          RETURN
0886        501 CONTINUE
0887          MARK=-2
```

```
FORTRAN IV     H01A-1    FRI 01-MAY-81 01:10:54          PAGE 003

0088          RETURN
0089     601  CONTINUE
0090          MARK=0
0091          RETURN
0092          END
```

Table 4

DFLIST.FOR

List of Data File	
Purpose: Reads the data file and presents the information on the CRT.	
Computer Output	**User Input**
	• Run DFLIST ⟨CR⟩
Pause -- Insert data disc in DK: and hit return	⟨CR⟩
Name input file	*DK:F20935.DAT ⟨CR⟩
Enter unit number for listing (6,7):[1]	6 ⟨CR⟩

[1] 6 = line printer
7 = teletype

Example Output - DFLIST

1	1	20035	20035	20035	20035	20035	20035	20035	20035	20035	20035	Subject No.
1	2	3	3	3	3	3	3	3	3	3	3	Session No.
1	3	0	0	0	0	0	0	0	0	0	0	
1	4	81	83	85	87	89	91	93	95	97	99	
1	5	-209	-213	-213	-208	-178	-159	-160	-146	-157	-158	Stick X Position
1	6	-212	-213	-214	-210	-180	-159	-158	-151	-153	-159	
1	7	-213	-214	-213	-210	-166	-159	-156	-146	-155	-158	
1	8	-212	-213	-212	-177	-159	-159	-156	-152	-157	-159	
1	9	334	334	333	333	337	333	356	358	341	343	Stick 4 Position
1	10	335	334	335	333	336	321	306	364	341	343	
1	11	335	336	334	331	337	314	301	361	341	340	
1	12	334	334	334	336	335	333	302	341	341	338	
1	13	-37	-37	-36	-36	-36	-36	-36	-36	-36	-38	Pedal Position
1	14	-37	-37	-36	-36	-36	-36	-36	-36	-36	-38	
1	15	559	697	775	775	773	615	592	598	266	224	Pitch Angle
1	16	637	745	784	753	667	593	555	426	237	171	
1	17	-568	-958	-1327	-1684	-1834	-1856	-2001	-2100	-2190	-2465	Roll Angle
1	18	-758	-1141	-1514	-1847	-1859	-1913	-2100	-2128	-2310	-2596	
1	19	1525	1561	1599	1637	1671	1701	1727	1749	1764	1769	Altitude
1	20	35	35	35	35	35	35	35	35	35	35	Torque
1	21	918	906	894	882	876	876	882	894	912	942	Airspeed
1	22	15542	15387	14918	14383	13730	13050	12303	11467	10713	9792	Heading
1	23	1012	1178	1244	1182	1039	880	749	657	313	12	Vertical Velocity
1	24	-45	-47	-50	-53	-60	-60	-58	-58	-56	-50	Yaw Angle
1	25	-2785	-2786	-2787	-2789	-2792	-2795	-2799	-2805	-2810	-2816	
1	26	0	0	0	0	0	0	0	0	0	0	
1	27	0	0	0	0	0	0	0	0	0	-128	
1	28	0	0	0	0	0	0	0	0	0	0	
1	29	0	0	0	0	0	0	0	0	0	0	

Program Listing STRPLT

```
FORTRAN IV     H01A-1    FRI 08-MAY-81 00:05:09              PAGE 001

      C PROGRAM: 'STRPLT.FOR'
      C PURPOSE IS TO PLOT C-50 VARIABLES ON LINE PRINTER
      C DATE: JULY 26, 1980
      C ARMY RESEARCH INSTITUTE, FORT RUCKER
0001        LOGICAL*1 LHEAD,LPCHAR,LYTIT,LYSCAL,LXSCAL,
           1          LYLINE,LXLINE,LVARC,IESCAP,LTY,LYVAL
0002        DIMENSION LPCHAR(100),IBDATA(29,10),LHEAD(60),
           1          LYSCAL(101),LYLINE(103),
           1          LVARC(16),IVNUM(16),LYTIT(100),
           1          LYVAL(104),INDEX(16),XMIN(16),XMAX(16),
           1          RANGE(16)
0003        DATA LXLINE/124*,LYLINE/103*45*,
           1     LYSCAL/124,9*32,124,9*32,124,9*32,124,9*32,
           1            124,9*32,124,9*32,124,9*32,124,9*32,
           1            124,9*32,124,9*32,124*,
           1     LYVAL*'0',9*32,'1','0',8*32,'2','0',8*32,
           1            '3','0',8*32,'4','0',8*32,'5','0',8*32,
           1            '6','0',8*32,'7','0',8*32,'8','0',8*32,
           1            '9','0',8*32,'1','0','0',' ',
           1     LYTIT/43*32,'V',' ','A','X','I','S',' ','P','E',
           1            'R','C','E','N','T',43*32*,
           1     LVARC/82,66,89,112,114,97,
           1            94,118,104,122,121,5*42*,
           1     IESCAP/27*,LTY/121*,IVNUM/16*1*,
           1     INDEX/5,9,13,15,17,19,20,21,22,23,24,5*0*
0004        PAUSE 'READY LINE PRINTER AND HIT CONTINUE'
0005        PAUSE 'INSERT DATA DISC IN UNIT ONE (DX:) AND HIT RETURN'
0006        WRITE(7,7000)
0007        CALL ASSIGN(2,     -1,'DX','NC',1)
0008        WRITE(7,7010)
0009        READ(5,5010) (LHEAD(I),I=1,60)
0010        WRITE(7,7030)
0011        READ(5,5030) (IVNUM(I),I=1,11)
0012        WRITE(7,7035)
0013        READ(5,5035) ISBJCT
0014        WRITE(7,7036)
0015        READ(5,5036) IUNIT
0016        WRITE(7,7037)
0017        READ(5,5037) IBEGIN
0018        CALL SCALE(IBDATA,2,IBEGIN,XMAX,XMIN,RANGE,INDEX,ISBJCT)
0019        IF(XMAX.EQ.0) GO TO 999
0021        WRITE(IUNIT,6010)
0022        WRITE(IUNIT,6020) (LHEAD(I),I=1,60)
0023        WRITE(IUNIT,6030)
0024        DO 10 I=1,11
0025        IF(IVNUM(I).NE.1) GO TO 10
0027        WRITE(IUNIT,6040) I,INDEX(I),XMIN(I),XMAX(I),RANGE(I)
0028       10 CONTINUE
0029        WRITE(IUNIT,6050) (I,LVARC(I),I=1,11)
0030        NBLOCK=0
0031        IXLOC=0
0032        ISPANT=0
0033       100 CONTINUE
0034        CALL INPUT(IBDATA,IBEGIN,XMAX,XMIN,2)
```

```
FORTRAN IV     H01A-1    FRI 08-MAY-81 08:05:09          PAGE 002

0035          IF(MARK.EQ.0) GO TO 800
0037          IF(MARK.EQ.-2) GO TO 810
0039          IF(NBLOCK.NE.1) GO TO 120
0041          WRITE(IUNIT,6410)
0042          DO 115 I=1,10
0043          IF(IBDATA(1,I).NE.ISBJCT) GO TO 115
0045          DO 110 J=1,11
0046          ISUB=INDEX(J)
0047          WRITE(IUNIT,6400) J,ISUB,IBDATA(ISUB,I)
0048      110 CONTINUE
0049          GO TO 116
0050      115 CONTINUE
0051      116 CONTINUE
0052          WRITE(IUNIT,6061)
0053          IF(IUNIT.EQ.7) GO TO 120
0055          WRITE(6,6050) IESCAP,LTV
0056          WRITE(6,6070) (LHEAD(I),I=1,60)
0057          WRITE(6,6080) (LYTIT(I),I=1,100)
0058          WRITE(6,6090) (LYVAL(I),I=1,104)
0059          WRITE(6,6100) (LYSCAL(I),I=1,101)
0060          WRITE(6,6110) (LYLINE(I),I=1,103)
0061      120 CONTINUE
0062          DO 300 I=1,10
0063          IF(IBDATA(1,I).NE.ISBJCT) GO TO 300
0065          ISPRNT=ISPRNT+1
0066          IXLOC=IXLOC+1
0067          DO 220 J=1,100
0068          LPCHAR(J)=32
0069      220 CONTINUE
0070          DO 260 K=1,11
0071          IF(IVNUM(K).NE.1) GO TO 260
0073          ISUB=INDEX(K)
0074          IVALUE=IBDATA(ISUB,I)
0075          IF(ISUB.EQ.22) IVALUE=IABS(IVALUE)
0077          RVALUE=ABS(FLOAT(IVALUE)-XMIN(K)+1.0)
0078          YLOC=RVALUE/RANGE(K)*100.0+0.5
0079          IF(IUNIT.EQ.7) YLOC=YLOC/2.0
0081          IYLOC=IFIX(YLOC)
0082          IF(IYLOC.GT.100) IYLOC=100
0084          IF((IUNIT.EQ.7).AND.(IYLOC.GT.50)) IYLOC=50
0085          IF(IYLOC.LT.1) IYLOC=1
0088          IF(IXLOC.LT.IBEGIN) GO TO 230
0090          IF(IUNIT.EQ.7) GO TO 230
0092          WRITE(7,7300) NBLOCK,I,K,IBDATA(4,I),IVALUE,IXLOC,IYLOC
0093      230 CONTINUE
0094          IF(LPCHAR(IYLOC).EQ.32) LPCHAR(IYLOC)=LVARC(K)
0096          IF(ISUB.NE.22) GO TO 260
0098          IF((IHSAVE.LT.0).AND.(IBDATA(ISUB,I).GE.0))
             1 LPCHAR(IYLOC)=43
0100          IF((IHSAVE.GE.0).AND.(IBDATA(ISUB,I).LT.0))
             1 LPCHAR(IYLOC)=45
0102          IHSAVE=IBDATA(ISUB,I)
0103      260 CONTINUE
0104          LYSCAL=32
```

Program Listing STRPLT (Continued)

```
FORTRAN IV     H01A-1    FRI 08-MAY-81 00:05:09              PAGE 003

0105          IF(ISPRNT.EQ.10) LXSCAL=45
0107          IF(ISPRNT.EQ.10) ISPRNT=0
0109          IF(IXLOC.LT.IBEGIN) GO TO 300
0111          IF(IUNIT.EQ.7) GO TO 270
0113          WRITE(6,6120) IBDATA(3,I),IBDATA(2,I),IBDATA(4,I),
             1              LXSCAL,LXLINE,(LPCHAR(IJK),IJK=1,100),
             1              LXLINE,LXSCAL,IBDATA(4,I),IXLOC
0114          GO TO 300
0115     270 CONTINUE
0116          WRITE(7,7120) IBDATA(3,I),IBDATA(2,I),IBDATA(4,I),IXLOC,
             1              LXSCAL,LXLINE,(LPCHAR(IJK),IJK=1,50),
             1              LXLINE,LXSCAL
0117     300 CONTINUE
0118          GO TO 100
0119     800 CONTINUE
0120          WRITE(6,7100)
0121          GO TO 820
0122     810 CONTINUE
0123          IF(IUNIT.EQ.7) GO TO 820
0125          WRITE(6,6110) (LVLINE(I),I=1,103)
0126          WRITE(6,6100) (LVSCAL(I),I=1,101)
0127          WRITE(6,6090) (LVVAL(I),I=1,104)
0128          WRITE(6,6091) (LVTIT(I),I=1,100)
0129          WRITE(6,6070) (LHEAD(I),I=1,60)
0130     820 CONTINUE
0131          CALL CLOSE(2)
0132          STOP 'END OF PROGRAM STRPLT.FOR'
0133    5010 FORMAT(60A1)
0134    5030 FORMAT(1117)
0135    5035 FORMAT(I7)
0136    5036 FORMAT(I7)
0137    5037 FORMAT(I7)
0138    7090 FORMAT(' NAME THE INPUT FILE (ASSIGN TO DK:)'/)
0139    7010 FORMAT(' ENTER TITLE FOR PLOT (60A1 MAX)')
0140    7030 FORMAT(' ENTER FLAG FOR THE ELEVEN VARIABLES (1117)')
0141    7035 FORMAT(' ENTER STUDENT NUMBER (I)')
0142    7036 FORMAT(' ENTER OUTPUT UNIT NUMBER (LP:=6,TT:=7)')
0143    7037 FORMAT(' ENTER BEGINNING LOCATION OF PLOT (I)')
0144    7100 FORMAT(' READ ERROR IN INPUT FILE')
0145    7300 FORMAT(1X,717)
0146    6010 FORMAT(' ',5X,'C-50 STRIP PLOTTER ROUTINE'/
             1 6X,'ARMY RESEARCH INSTITUTE, FORT RUCKER FIELD UNIT'/
             1 6X,'MULTIVARIATE ANALYSIS OF TIME SEQUENCED DATA'/
             1 6X,'UH-1 (HUEY) STUDENT PILOT SIMULATOR SESSIONS'/' ')
0147    6020 FORMAT(' ',5X,'TITLE: ',60A1)
0148    6030 FORMAT(' '/' ',5X,'VARIABLE SCALES:'/' '/
             1 ' ',5X,'INDEX',3X,'SUB',
             1 2X,'MINIMUM',2X,'MAXIMUM',6X,'RANGE'/
             1 ' ',5X,'-----',3X,'---',
             1 2X,'---------',2X,'---------',6X,'-----'/' ')
0149    6040 FORMAT(6X,I5,2X,I4,2X,F7.0,2X,F7.0,2X,F9.0)
0150    6050 FORMAT(1X,2A1,'1')
0151    6060 FORMAT(' '/' '/
             1 ' ',5X,'VARIABLE KEYS:'/' '/
```

Program Listing STRPLT (Concluded)

```
FORTRAN IV     HQ1A-1    FRI 09-MAY-81 00:05:09          PAGE 004

          1 ' ',5X,'INDEX',2X,'CHAR',2X,'DESCRIPTION'/
          1 ' ',5X,'-----',2X,'----',2X,'-----------'/ ' '/
          1 6X,I5,2X,A1,5X,'STICK X (ROLL INPUT)'/
          1 6X,I5,2X,A1,5X,'STICK Y (PITCH INPUT)'/
          1 6X,I5,2X,A1,5X,'PEDAL POSITION'/
          1 6X,I5,2X,A1,5X,'PITCH'/
          1 6X,I5,2X,A1,5X,'ROLL'/
          1 6X,I5,2X,A1,5X,'ALTITUDE'/
          1 6X,I5,2X,A1,5X,'TORQUE'/
          1 6X,I5,2X,A1,5X,'AIRSPEED'/
          1 6X,I5,2X,A1,5X,'HEADING'/
          1 6X,I5,2X,A1,5X,'VERTICAL VELOCITY'/
          1 6X,I5,2X,A1,5X,'YAW'/' ')
 0152  6051 FORMAT(1X/6X,'NOTE 1: ALL SCALES RELATIVE EXCEPT HEADING'/
          1 6X,'NOTE 2: VARIABLES SAMPLED ONCE EVERY TWO SECONDS'/
          1 6X,'NOTE 3: SYMBOL "+" INDICATES HEADING HAS',
          1 ' CHANGED FROM NEGATIVE TO POSITIVE'/
          1 6X,'NOTE 4: SYMBOL "-" INDICATES HEADING HAS',
          1 ' CHANGED FROM POSITIVE TO NEGATIVE'/' ')
 0153  6070 FORMAT(18X,60A1/1X/1X)
 0154  6090 FORMAT(18X,100A1/1X)
 0155  6091 FORMAT(1X/18X,100A1/1X)
 0156  6090 FORMAT(17X,104A1)
 0157  6100 FORMAT(17X,101A1)
 0158  6110 FORMAT(16X,103A1)
 0159  6120 FORMAT(4X,I2,I5,I4,1X,104A1,I4,I6)
 0160  7120 FORMAT(1X,I3,1X,I5,1X,I5,1X,I5,4X,54A1)
 0161  6410 FORMAT(6X,'INTIAL VALUES:'/1X/
          1 6X,'INDEX',2X,'SUB',4X,'VALUE'/
          1 6X,'-----',2X,'---',4X,'-----'/' ')
 0162  6400 FORMAT(6X,I5,2X,I3,2X,I7)
 0163       END
```

Subroutine SCALE

```
FORTRAN IV     H01A-1    FRI 09-MAY-81 00:05:37              PAGE 001

0001          SUBROUTINE SCALE(IBDATA,XMIN,XMAX,MAPX,PAGE,INDEX,ISBJCT)
0002          DIMENSION IBDATA(29,10),XMIN(16),XMAX(16),
             1          MIN(16),MAX(16),RANGE(16),INDEX(16)
0003          NBLOCK=0
0004          DO 10 I=1,16
0005          MIN(I)=32767
0006          MAX(I)=-32767
0007       10 CONTINUE
0008      100 CONTINUE
0009          CALL INPUT(IBDATA,NBLOCK,MAPX,2)
0010          IF(MAPX.NE.10) GO TO 400
0012          DO 200 I=1,10
0013          IF(IBDATA(1,I).NE.ISBJCT) GO TO 200
0015          DO 160 K=1,11
0016          ISUB=INDEX(K)
0017          IF(ISUB.EQ.22) IBDATA(ISUB,I)=IABS(IBDATA(ISUB,I))
0019          IF(IBDATA(ISUB,I).LT.MIN(K)) MIN(K)=IBDATA(ISUB,I)
0021          IF(IBDATA(ISUB,I).GT.MAX(K)) MAX(K)=IBDATA(ISUB,I)
0023      160 CONTINUE
0024      200 CONTINUE
0025          GO TO 100
0026      400 CONTINUE
0027          WRITE(7,7010)
0028          DO 500 I=1,11
0029          WRITE(7,7020) I,MIN(I),MAX(I)
0030      500 CONTINUE
0031          DO 600 I=1,16
0032          XMIN(I)=FLOAT(MIN(I))
0033          XMAX(I)=FLOAT(MAX(I))
0034          IF((MIN(I).LT.0).AND.(MAX(I).LT.0))
             1 RANGE(I)=ABS(XMIN(I))-ABS(XMAX(I))+1.0
0036          IF((MIN(I).LT.0).AND.(MAX(I).GE.0))
             1 RANGE(I)=ABS(XMIN(I))+XMAX(I)+1.0
0038          IF((MIN(I).GE.0).AND.(MAX(I).LT.0))
             1 RANGE(I)=ABS(XMAX(I))+XMIN(I)+1.0
0040          IF((MIN(I).GE.0).AND.(MAX(I).GE.0))
             1 RANGE(I)=XMAX(I)-XMIN(I)+1.0
0042      600 CONTINUE
0043          XMIN(9)=0.0
0044          XMAX(9)=18000.0
0045          RANGE(9)=18001.0
0046          REWIND 2
0047          RETURN
0048      7010 FORMAT(' VARIABLE MINIMUM AND MAXIMUM VALUES')
0049      7020 FORMAT(1X,I5,2X,I7,2X,I7)
0050          END
```

Subroutine INPUT

```
FORTRAN IV     V01A-1    FRI 09-MAY-81 08:05:53              PAGE 001

0001          SUBROUTINE INPUT(X,NBLOCK,MARK,N)
0002          INTEGER DATA,X
0003          DIMENSION DATA(256),X(29,10)
0004          IP=0
0005          IPL=0
0006          DO 10 J=1,10
0007          DO 10 I=1,29
0008          X(I,J)=0
0009       10 CONTINUE
0010          MARK=0
0011          READ(N,END=501,ERR=601) IDUM1,IDUM2,IDUM3,IDUM4,IDUM5,IDUM6,
             1                        (DATA(I),I=1,250)
0012          NBLOCK=NBLOCK+1
0013          K=0
          C MAIN LOOP
0014          DO 100 J=1,10
0015          K=K+2
          C STUDENT NUMBER
0016          X(1,J)=DATA(K)
          C PERIOD
0017          K=K+2
0018          X(2,J)=MOD(DATA(K),256)
          C SEGMENT NUMBER
0019          K=K+10
0020          DO 20 I=1,2
0021          K=K+1
0022          HTY=MOD(DATA(K),2)
0023          X(3,J)=X(3,J)+HTY
0024          DATA(K)=DATA(K)-HTY
0025       20 CONTINUE
          C TIME
0026          K=K-12
0027       30 LPS=DATA(K)/256+IP
0028          IF(IPL.LT.LPS) GO TO 40
0029          IP=IP+256
0031          GO TO 30
0032       40 IPL=LPS
0033          X(4,J)=LPS
          C STICK X (ROLL INPUT) SCALING IS INCHES TIMES 100 MAX=6.24"
0034          DO 50 I=5,8
0035          K=K+1
0036          Y=FLOAT(DATA(K))*0.019044+ISIGN(5,DATA(K))/10.0
0037          X(I,J)=INT(Y)
0038       50 CONTINUE
          C STICK Y (PITCH INPUT) SCALING IS INCHES TIMES 100 MAX=6.333"
0039          DO 60 I=9,12
0040          K=K+1
0041          Y=FLOAT(DATA(K))*0.019323+ISIGN(5,DATA(K))/10.0
0042          X(I,J)=INT(Y)
0043       60 CONTINUE
0044          DO 70 I=13,14
0045          K=K+1
          C SIDE TASK START FLAG
0046          HTY=MOD(DATA(K),2)
```

```
FORTRAN IV     M01A-1    FRI 08-MAY-81 08:05:53              PAGE 002

0047          DATA(K)=DATA(K)-NTY
0048          X(26,J)=X(26,J)+100*NTY
      C PEDAL POSTION SCALING IS INCHES TIMES 100 MAX=3.25"
0049          Y=FLOAT(DATA(K))*0.009919+ISIGN(5,DATA(K))/10.0
0050          X(1,J)=INT(Y)
0051       70 CONTINUE
      C PITCH INDICATED MAX=60 DEGREES, SCALED TIMES 100
0052          DO 80 I=15,16
0053          K=K+1
0054          DATA(K)=DATA(K)-MOD(DATA(K),2)
0055          Y=FLOAT(DATA(K))*0.54935+ISIGN(5,DATA(K))/10.0
0056          X(I,J)=INT(Y)
0057       80 CONTINUE
      C ROLL INDICATED MAX=90 DEGREES, SCALED TIMES 100
0058          DO 90 I=17,18
0059          K=K+1
0060          DATA(K)=DATA(K)-MOD(DATA(K),2)
0061          Y=FLOAT(DATA(K))*0.54932+ISIGN(5,DATA(K))/10.0
0062          X(I,J)=INT(Y)
0063       90 CONTINUE
      C ALTITUDE
0064          K=K+1
0065          X(19,J)=DATA(K)
      C TORQUE   0 TO 100 PSI
0066          K=K+1
0067          X(20,J)=DATA(K)/256
      C AIRSPEED MAX=146.5 KNOTS, SCALED TIMES 100
0068          X(21,J)=64*MOD(DATA(K),256)
      C HEADING 0 TO 360 DEGREES, SCALED TIMES 100
0069          K=K+1
0070          X(22,J)=FLOAT(DATA(K))*0.54932+ISIGN(5,DATA(K))/10.0
      C VERTICAL VELOCITY
0071          K=K+1
0072          X(23,J)=DATA(K)
      C YAW 0 TO 14.9 DEGREES, SCALED TIMES 100
0073          K=K+1
0074          X(24,J)=FLOAT(DATA(K))*0.04548+ISIGN(5,DATA(K))/100.0
      C COURSE DEVIATION
0075          K=K+1
0076          X(25,J)=FLOAT(DATA(K))*0.54932+ISIGN(5,DATA(K))/10.0
      C SIDE TASK NUMBER
0077          K=K-21
0078          X(26,J)=X(26,J)+MOD(DATA(K),256)
      C GENERATOR RECYCLE TIME
0079          X(29,J)=DATA(K)/256
      C STATION SELECT TIME
0080          K=K+22
0081          X(27,J)=DATA(K)/256
      C MIKE RESPONSE TIME
0082          X(28,J)=MOD(DATA(K),256)
0083       100 CONTINUE
0084          MARK=10
0085          RETURN
0086       501 CONTINUE
```

```
IV    H01A-1   FRI 00-MAY-81 00:05:53          PAGE 003

      MARK:=-2
      RETURN
601   CONTINUE
      MARK=0
      RETURN
      END
```

Table 5

STRPLT.FOR

Strip Plot	
Purpose: To plot flight variables on the line printer.	
Computer Output	User Input
	• Run STRPLT
Pause -- Ready line printer and hit continue	⟨CR⟩
Pause -- Insert data disc in unit one (DK:) and hit return	⟨CR⟩
Name the input file (assign to DK:)	*DK: F17932. DAT ⟨CR⟩
Enter title for plot (goal Max)	Subject 17932 ⟨CR⟩
Enter flag for the eleven variables (11I7)	0,0,0,1,1,1,0,1,1,0,0[1] ⟨CR⟩
Enter student number (I)	17932 ⟨CR⟩

0 = Unwanted variables
1 = Desired variables

Table 5

STRPLT.FOR (Concluded)

Strip Plot	
Computer Output	Computer Input
Enter output unit number (LP: = 6[2], TT: = 7)[3]	6 ⟨CR⟩
Enter beginning location of plot	0 ⟨CR⟩
[2]LP = Line printer [3]TT = teletype	

C-50 STRIP PLOTTER ROUTINE
ARMY RESEARCH INSTITUTE, FORT RUCKER FIELD UNIT
MULTIVARIATE ANALYSIS OF TIME SEQUENCED DATA
UH-1 (HUEY) STUDENT PILOT SIMULATOR SESSIONS

TITLE: SUBJECT 9933

VARIABLE SCALES:

INDEX	SUB	MINIMUM	MAXIMUM	RANGE
4	15	-385.	1665.	1971.
5	17	-1581.	3231.	4813.
6	19	991.	1496.	506.
8	21	768.	978.	211.
9	22	0.	18000.	18001.

VARIABLE KEYS:

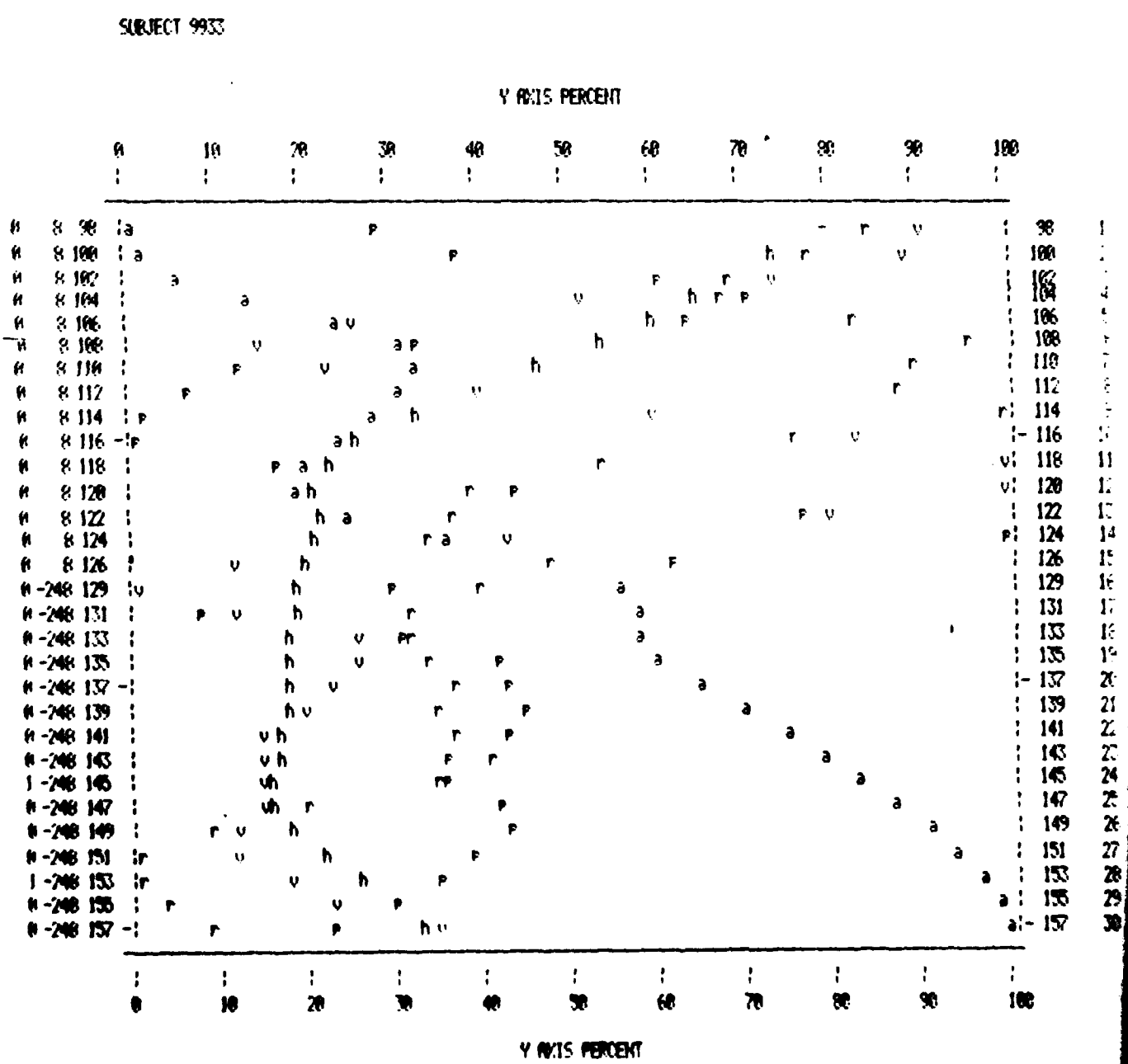
INDEX	CHAR	DESCRIPTION
1	R	STICK X (ROLL INPUT)
2	P	STICK Y (PITCH INPUT)
3	Y	PEDAL POSITON
4	p	PITCH
5	r	ROLL
6	a	ALTITUDE
7	T	TORQUE
8	v	AIRSPEED
9	h	HEADING
10	z	VERTICAL VELOCITY
11	y	YAW

INTIAL VALUES:

INDEX	SUB	VALUE
1	5	-112
2	9	246
3	13	-38
4	15	271
5	17	2485
6	19	991
7	20	28
8	21	960
9	22	-14421
10	23	210
11	24	10

NOTE 1: ALL SCALES RELATIVE EXCEPT HEADING
NOTE 2: VARIABLES SAMPLED ONCE EVERY TWO SECONDS
NOTE 3: SYMBOL "+" INDICATES HEADING HAS CHANGED FROM NEGATIVE TO POSITIVE
NOTE 4: SYMBOL "-" INDICATES HEADING HAS CHANGED FROM POSITIVE TO NEGATIVE

Example Output STRPLT (Concluded)



Program Listing PHAN 3

```
FORTRAN IV     V01A-1    THU 07-MAY-81 08:52:19              PAGE 001

      C     PROGRAM: PHAN3.FOR
      C     PURPOSE: RETURNS CELL VALUES FOR C-50 FLIGHT DATA
      C     CONTRACT: C-50
      C     DATE: 21-FEB-81
      C
0001        COMMON VARBLE(3000),ERROR(3000)
0002        DIMENSION ICELLS(25),ITMATX(25,25),KSTART(100),
           1KSTOP(100),AMEAN(4,100)
0003        INTEGER VARBLE,SUBJCT,NUMVAR,NUM,SUB1,SUB2,SUB3,DEV,CELL,CELLM1
0004        REAL I,J,CHKVAL,ERMEAN
0005        DATA TOTERR,TOTSQR,ERRSAR,ERMEAN/0.0,0.0,0.0,0.0/
0006        DATA RATERR,TOTRTE,RTESQR,ERRSQR/0.0,0.0,0.0,0.0/
0007        DATA SUB1,SUB2,SUB3,NUMVAR/0,0,0,0/
0008        DATA ICELLS/25*0/,ITMATX/625*0/
      C
0009     17 CONTINUE
0010        PAUSE' HIT RETURN & NAME OUTPUT FILE'
0011        CALL ASSIGN(1,' ',-1,'NEW','NC',1)
0012        PAUSE' INSERT DATA DISC IN DRIVE ONE AND HIT RETURN'
0013        PAUSE' HIT RETURN AND NAME THE 1ST INPUT FILE'
0014        CALL ASSIGN(2,' ',-1,'RDO','NC',1)
0015        WRITE(7,7022)
0016        READ(5,5002)SUBJCT
0017        WRITE(7,7000)
0018        READ(5,5000)CHKVAL
0019        WRITE(7,7023)
0020        READ(5,5023)ERMEAN,ERRDEV,RTMEAN,RTEDEV
0021        WRITE(7,7010)
0022        READ(5,5010) NUM
0023        IF(NUM.EQ.21) SCALE=10.0
0024        IF(NUM.EQ.19) SCALE=1.0
0025        IF(NUM.EQ.22) SCALE=100.0
0026        IF(NUM.EQ.23) SCALE=1.0
0027        WRITE(7,7020)
0028        READ(5,5020) DEV
0029     10    DO 21 II=1,100
0030        READ(2,5021)KSTART(II),KSTOP(II),(AMEAN(JJ,II),JJ=1,4)
0031        IF(KSTART(II).LE.0)GO TO 22
0032     21 CONTINUE
0033     22 ILIST=II-1
0034        CALL CLOSE(2)
0035        PAUSE'HIT RETURN & NAME 2ND INPUT FILE'
0036        CALL ASSIGN(2,' ',-1,'RDO','NC',1)
0037        DO 23 II=1,ILIST
0038        IF(NUM.EQ.19) ERMEAN=AMEAN(1,II)
0039        IF(NUM.EQ.21) ERMEAN=AMEAN(2,II)
0040        IF(NUM.EQ.22) ERMEAN=AMEAN(3,II)
0041        IF(NUM.EQ.23) ERMEAN=AMEAN(4,II)
0042        ISTART=KSTART(II)
0043        ISTOP=KSTOP(II)
0044        CALL READ1(NUM,NUMVAR,ISTART,ISTOP,SUBJCT)
      C
0045        DO 20 K=1,NUMVAR
0046        ERR=(FLOAT(VARBLE(K))/SCALE)-CHKVAL-ERMEAN
```

```
FORTRAN IV      V01A-1    THU 07-NOV-81 08:52:19              PAGE 002

0056          ERROR(K)=ERR
0057    20    CONTINUE
0058          NUMR1=NUMVAR-1
      C
0059          WRITE(DEV,7050) SUBJCT,NUM,
             1 NUMVAR,SCALE,ISTART,ISTOP,ERMEAN,ERRDEV,RTMEAN,RTEDEV
0060          WRITE(DEV,7051)
      C
0061          A=ERMEAN+(1.5*ERRDEV)
0062          B=ERMEAN+(ERRDEV*0.5)
0063          C=ERMEAN-(ERRDEV*0.5)
0064          D=ERMEAN-(1.5*ERRDEV)
0065          E=RTMEAN+(1.5*RTEDEV)
0066          F=RTMEAN+(RTEDEV*0.5)
0067          G=RTMEAN-(RTEDEV*0.5)
0068          H=RTMEAN-(1.5*RTEDEV)
      C
0069          WRITE(DEV,7051)
0070          WRITE(DEV,7070)
0071          WRITE(DEV,7051)
      C
0072          LINES=15
0073          DO 40 N=1,NUMR1
0074             I=ERROR(N)
0075             J=ERROR(N+1)-ERROR(N)
0076             CELL=0
0077             IF(J.GE.E.AND.I.GE.A) CELL=5
0079             IF(J.GE.E.AND.I.GE.B.AND.I.LT.A) CELL=4
0081             IF(J.GE.E.AND.I.GE.C.AND.I.LT.B) CELL=3
0083             IF(J.GE.E.AND.I.GE.D.AND.I.LT.C) CELL=2
0085             IF(J.GE.E.AND.I.LT.D) CELL=1
0087             IF(J.GE.F.AND.J.LT.E.AND.I.GE.A) CELL=10
0089             IF(J.GE.F.AND.J.LT.E.AND.I.GE.B.AND.I.LT.A) CELL=9
0091             IF(J.GE.F.AND.J.LT.E.AND.I.GE.C.AND.I.LT.B) CELL=8
0093             IF(J.GE.F.AND.J.LT.E.AND.I.GE.D.AND.I.LT.C) CELL=7
0095             IF(J.GE.F.AND.J.LT.E.AND.I.LT.D) CELL=6
0097             IF(J.GE.G.AND.J.LT.F.AND.I.GE.A) CELL=15
0099             IF(J.GE.G.AND.J.LT.F.AND.I.GE.B.AND.I.LT.A) CELL=14
0101             IF(J.GE.G.AND.J.LT.F.AND.I.GE.C.AND.I.LT.B) CELL=13
0103             IF(J.GE.G.AND.J.LT.F.AND.I.GE.D.AND.I.LT.C) CELL=12
0105             IF(J.GE.G.AND.J.LT.F.AND.I.LT.D) CELL=11
0107             IF(J.GE.H.AND.J.LT.G.AND.I.GE.A) CELL=20
0109             IF(J.GE.H.AND.J.LT.G.AND.I.GE.B.AND.I.LT.A) CELL=19
0111             IF(J.GE.H.AND.J.LT.G.AND.I.GE.C.AND.I.LT.B) CELL=18
0113             IF(J.GE.H.AND.J.LT.G.AND.I.GE.D.AND.I.LT.C) CELL=17
0115             IF(J.GE.H.AND.J.LT.G.AND.I.LT.D) CELL=16
0117             IF(J.LT.H.AND.I.GE.A) CELL=25
0119             IF(J.LT.H.AND.I.GE.B.AND.I.LT.A) CELL=24
0121             IF(J.LT.H.AND.I.GE.C.AND.I.LT.B) CELL=23
0123             IF(J.LT.H.AND.I.GE.D.AND.I.LT.C) CELL=22
0125             IF(J.LT.H.AND.I.LT.D) CELL=21
0127             IF(CELL.EQ.0) PAUSE 'ERROR IN CELL DETERMINATION'
0129             WRITE(DEV,7060) N,I,J,CELL
0131             ICELLS(CELL)=ICELLS(CELL)+1
```

Program Listing PHAN 3 (Continued)

```
FORTRAN IV      HW1A-1    THU 07-MAY-81 08:52:19              PAGE 005

0131            IF(N.LT.2) CELLM1=CELL
0133            IF(N.LT.2) GO TO 39
0135            ITMATX(CELLM1,CELL)=ITMATX(CELLM1,CELL)+1
0136            CELLM1=CELL
0137    39      CONTINUE
0138            LINES=LINES+1
0139            IF(LINES.LT.45) GOTO 40
0141              LINES=0
0142              WRITE(DEV,7071)
0143    40    CONTINUE
0144          REWIND 2
0145    73    CONTINUE
0146    11    WRITE(DEV,7090) (ICELLS(M),M=1,25)
0147          WRITE(DEV,7091) ((ITMATX(M,N),N=1,25),M=1,25)
0148          DO 80 M=1,25
0149    80    WRITE(1,5024)(ITMATX(M,N),N=1,25)
0150          CALL CLOSE(1)
0151          CALL CLOSE(DEV)
0152          STOP
          C
0153    5000  FORMAT(F13.7)
0154    5010 FORMAT(I6)
0155    5011 FORMAT(F13.7)
0156    5021 FORMAT(2I7,4F8.0)
0157    5020 FORMAT(I6)
0158    5027 FORMAT(I7)
0159    5023 FORMAT(4F15.4)
0160    5024 FORMAT(25I4)
0161    7000 FORMAT(' ENTER CHECK VALUE TO BE ANALYSED'/
               1       ' AIRSPEED-ENTER 90.00'/
               1       ' ALTITUDE-ENTER 2000.00'/
               1       ' HEADING -ENTER 90.00'/
               1       ' RATE OF CLIMB-ENTER 0.00'/)
0162    7010 FORMAT(' FOR THE VARIABLE TO BE ANALYZED:'/
               1       ' AIRSPEED - ENTER 21'/
               1       ' ALTITUDE - ENTER 19'/
               1       ' HEADING - ENTER 22'/
               1       ' RATE OF CLIMB - ENTER 23'/)
0163    7020 FORMAT(' FOR OUTPUT 6=LP 7=TT ')
0164    7021 FORMAT(' ENTER THE START AND STOP POSITION IN FILE, ENTER ERMEAN')
0165    7022 FORMAT(' ENTER SUBJECT NUMBER')
0166    7023 FORMAT('ENTER ERMEAN,ERRDEV,RTMEAN,RTEDEV')
0167    7050 FORMAT(' SUBJECT: ',I6/
               1       ' VARIABLE: ',I2/
               1       ' NUMBER OF SAMPLES: ',I5/
               1       ' VARIABLE SCALE: ',F10.5/
               1       ' START POSITION: ',I5/
               1       ' STOP POSTION: ',I5/
               1       ' MEAN OF ERROR = ',F14.4/
               1       ' ERROR DEVIATION = ',F14.4/
               1       ' MEAN OF ERROR RATE = ',F14.4/
               1       ' DEVIATION OF ERROR RATE = ',F14.4/)
0168    7051 FORMAT(' ')
0169    7060 FORMAT('   CELL VALUES '//
```

```
      1      '  ----+----+----+----+----  '/
      1      '   1    2    3    4    5    '/
      1      '  ----+----+----+----+----  +1.5
      1      '   6    7    8    9   10
      1      '  ----+----+----+----+----
      1      '  11   12   13   14   15     RATE OF ERROR
      1      '  ----+----+----+----+----
      1      '  16   17   18   19   20
      1      '  ----+----+----+----+----
      1      '  21   22   23   24   25
      1      '  ----+----+----+----+----  '
      1      '     -1.5  -.5  +.5  +1.5
      1      '              ERROR'/)
 7070 FORMAT(' ','INDEX',5X,'ERROR',2X,'RATE OF ERROR',2X,'CELL'/)
 7071 FORMAT('1','INDEX',5X,'ERROR',2X,'RATE OF ERROR',2X,'CELL'/)
 7000 FORMAT(' ',I5,1X,F9.3,2X,F13.3,2X,I4)
 7090 FORMAT('1'//'   CELL VALUES '//
      1      '  ----+----+----+----+----  '/
      1 ' ',1X,5(1X,I5,1X)/
      1      '  ----+----+----+----+----  +1.5  '
      1 ' ',1X,5(1X,I5,1X)/
      1      '  ----+----+----+----+----  +.5 '/
      1 ' ',1X,5(1X,I5,1X),'   RATE OF ERROR'/
      1      '  ----+----+----+----+----  -.5 '/
      1 ' ',1X,5(1X,I5,1X)/
      1      '  ----+----+----+----+----  -1.5 '/
      1 ' ',1X,5(1X,I5,1X)/
      1      '  ----+----+----+----+----  '/
      1      '     -1.5  -.5  +.5  +1.5     '/
      1      '              ERROR'/)
 7091 FORMAT('1'/
      1 ' ','CELL TRANSTION MATRIX:'/
      1 25(' ',25I4/))
      END
```

Subroutine READ

```
                    H01A-1    MON 04-MAY-81 02:38:02              PAGE 001

      1         SUBROUTINE READ1(NUM,NUMAP,ISTART,ISTOP,SUBJCT)
      2         COMMON APPLE(3000),EPROP(3000)
      3         DIMENSION X(29,10)
      4         INTEGER APPLE,X,SUBJCT
      5         ISUB=0
      6         NBLOCK=0
      7         NUMAP=0
      8   C
      9         DO 30 I=1,200
     10           CALL INPUT(X,NBLOCK,NAPX,2)
     11           IF(NAPX.EQ.-2) GO TO 31
     12           DO 20 J=1,10
     13             IF(X(1,J).NE.SUBJCT) GO TO 10
     14             ISUB=ISUB+1
     15             IF(ISUB.LT.ISTART) GO TO 10
     16             IF(ISUB.GT.ISTOP) GO TO 10
     17             NUMAP=NUMAP+1
     18             APPLE(NUMAP)=X(NUM,J)
     19     10      CONTINUE
     20     20    CONTINUE
     21     30  CONTINUE
     22     31  CONTINUE
     23         DO 40 K=1,NUMAP
     24           WRITE(7,7000) K,APPLE(K)
     25     40  CONTINUE
     26   C
     27   7000 FORMAT(' ',I5,1X,I6)
     28         END
```

```
FORTRAN IV     H01A-1    MON 04-MAY-81 02:38:15              PAGE 001

0001        SUBROUTINE INPUT(X,NBLOCK,MARK,N)
0002        INTEGER DATA,X
0003        DIMENSION DATA(250),X(29,10)
0004        IP=0
0005        IPL=0
0006        DO 10 J=1,10
0007        DO 10 I=1,29
0008        X(I,J)=0
0009     10 CONTINUE
0010        MARK=0
0011        READ(N,END=501,ERR=601) IDUM1,IDUM2,IDUM3,IDUM4,IDUM5,IDUM6,
           1                        (DATA(I),I=1,250)
0012        NBLOCK=NBLOCK+1
0013        K=0
      C MAIN LOOP
0014        DO 100 J=1,10
0015        K=K+2
      C STUDENT NUMBER
0016        X(1,J)=DATA(K)
      C PERIOD
0017        K=K+2
0018        X(2,J)=MOD(DATA(K),256)
      C SEGMENT NUMBER
0019        K=K+10
0020        DO 20 I=1,2
0021        K=K+1
0022        NTK=MOD(DATA(K),2)
0023        X(3,J)=X(3,J)+NTK
0024        DATA(K)=DATA(K)-NTK
0025     20 CONTINUE
0026        K=K-12
0027     30 LPS=DATA(K)/256+IP
0028        IF(IPL.LT.LPS) GO TO 40
0030        IP=IP+256
0031        GO TO 30
0032     40 IPL=LPS
0033        X(4,J)=LPS
      C STICK X (ROLL INPUT) SCALING IS INCHES TIMES 100 MAX=6.24"
0034        DO 50 I=5,8
0035        K=K+1
0036        Y=FLOAT(DATA(K))*0.019044+ISIGN(5,DATA(K))/10.0
0037        X(I,J)=INT(Y)
0038     50 CONTINUE
      C STICK Y (PITCH INPUT) SCALING IS INCHES TIMES 100 MAX=6.333"
0039        DO 60 I=9,12
0040        K=K+1
0041        Y=FLOAT(DATA(K))*0.019323+ISIGN(5,DATA(K))/10.0
0042        X(I,J)=INT(Y)
0043     60 CONTINUE
0044        DO 70 I=13,14
0045        K=K+1
      C SIDE TASK START FLAG
0046        NTK=MOD(DATA(K),2)
0047        DATA(K)=DATA(K)-NTK
```

```
FORTRAN IV     H01A-1    MON 04-MAY-81 02:38:15              PAGE 002

0048          X(26,J)=X(26,J)+100*NTK
      C PEDAL POSITION SCALING IS INCHES TIMES 100 MAX=3.25"
0049          Y=FLOAT(DATA(K))*0.009919+ISIGN(5,DATA(K))/10.0
0050          X(1,J)=INT(Y)
0051       70 CONTINUE
      C PITCH INDICATED MAX=60 DEGREES, SCALED TIMES 100
0052          DO 80 I=15,16
0053          K=K+1
0054          DATA(K)=DATA(K)-MOD(DATA(K),2)
0055          Y=FLOAT(DATA(K))*0.54935+ISIGN(5,DATA(K))/10.0
0056          X(I,J)=INT(Y)
0057       80 CONTINUE
      C ROLL INDICATED MAX=40 DEGREES, SCALED TIMES 100
0058          DO 90 I=17,18
0059          K=K+1
0060          DATA(K)=DATA(K)-MOD(DATA(K),2)
0061          Y=FLOAT(DATA(K))*0.54932+ISIGN(5,DATA(K))/10.0
0062          X(I,J)=INT(Y)
0063       90 CONTINUE
      C ALTITUDE
0064          K=K+1
0065          X(19,J)=DATA(K)
      C TORQUE   0 TO 100 PSI
0066          K=K+1
0067          X(20,J)=DATA(K)/256
      C AIRSPEED MAX=146.5 KNOTS, SCALED TIMES 100
0068          X(21,J)=6*MOD(DATA(K),256)
      C HEADING 0 TO 360 DEGREES, SCALED TIMES 100
0069          K=K+1
0070          X(22,J)=FLOAT(DATA(K))*0.54932+ISIGN(5,DATA(K))/10.0
      C VERTICAL VELOCITY
0071          K=K+1
0072          X(23,J)=DATA(K)
      C RPM 0 TO 14.9 DEGREES, SCALED TIMES 100
0073          K=K+1
0074          X(24,J)=FLOAT(DATA(K))*0.04548+ISIGN(5,DATA(K))/100.0
      C COURSE DEVIATION
0075          K=K+1
0076          X(25,J)=FLOAT(DATA(K))*0.54932+ISIGN(5,DATA(K))/10.0
      C SIDE TASK NUMBER
0077          K=K-21
0078          X(26,J)=X(26,J)+MOD(DATA(K),256)
      C GENERATOR RECYCLE TIME
0079          X(29,J)=DATA(K)/256
      C STATION SELECT TIME
0080          K=K+22
0081          X(27,J)=DATA(K)/256
      C MIKE RESPONSE TIME
0082          X(28,J)=MOD(DATA(K),256)
0083      100 CONTINUE
0084          MAPK=10
0085          RETURN
0086      501 CONTINUE
0087          REWIND 2
```

```
[illegible]    H31A-1    MON 04-MAY-81 02:38:15          PAGE 008

[illegible]          ISAV1=-2
[illegible]          RETURN
[illegible]    681 CONTINUE
[illegible]          MARK=0
[illegible]          RETURN
[illegible]          END
```

Table 6

PHAN3.FOR

Return Cell Values for Flight Data	
Purpose: The user enters error mean, error deviation, mean error rate, and the deviation of the error rate. The computer reads the data file and computes phase plane cells and outputs a count matrix.	
Computer Output	User Input
	• Run PHAN 3 〈CR〉
Pause -- Hit return and name output file	〈CR〉 *DK:TTEST1.DAT 〈CR〉
Pause -- Insert data disc in drive one and hit return	〈CR〉
Pause -- Hit return and name the input file	〈CR〉 *DK:TTEST2.DAT 〈CR〉
Enter subject number	20935 〈CR〉
Enter check value to be analyzed: airspeed - enter 90.00; altitude - enter 2000.00; heading - enter 90.00;rate of climb - enter 0.00	2000.00 〈CR〉

Table 6

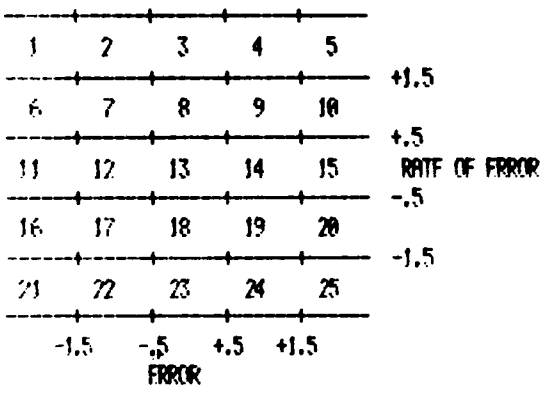
PHAN 3. FOR (Concluded)

Return Cell Values for Flight Data	
Computer Output	User Input
NTER ERMEAN, ERRDEV, RTMEAN, RTDEV[1]	-340.56, 84.43, 0., 8.22 ⟨CR⟩
For the variable to be analyzed: airspeed - enter 21; altitude - enter 19; heading - enter 22; rate of climb - enter 23	19 ⟨CR⟩
For output 6 = LP[2] 7= T[3]	6 ⟨CR⟩
Enter start and stop position in file, enter ERMEAN	1, 9, -340.56 ⟨CR⟩
	62, 73, 109.25 ⟨CR⟩
[1]NTER ERMEAN = Enter error mean ERRDEV = Error Deviation RTMEAN = Mean Error Rate RTDEV = Deviation of Error Rate	
[2]LP = Line Printer [3]TT = Teletype	
Note: To end enter a negative number for the start position. You may enter more than one set of START, STOP, ERMEANS at a time. Remember to ⟨CR⟩ between	

Example Output PHAN 3

```
SUBJECT: 20035
VARIABLE: 19
NUMBER OF SAMPLES:     9
VARIABLE SCALE:    1.00000
START POSITION:     1
STOP POSITION:     9
MEAN OF ERROR =      -340.5600
ERROR DEVIATION =        84.4300
MEAN OF ERROR RATE =         0.0000
DEVIATION OF ERROR RATE =         8.2200
```

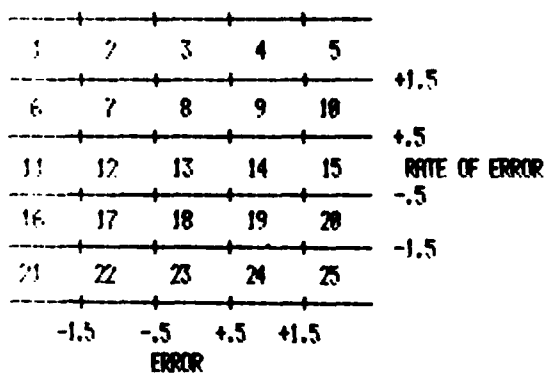
CELL VALUES



INDEX	ERROR	RATE OF ERROR	CELL
1	-133.440	35.000	5
2	-98.440	38.000	5
3	-60.440	38.000	5
4	-22.440	34.000	5
5	11.560	30.000	5
6	41.560	26.000	5
7	67.560	22.000	5
8	89.560	15.000	5

```
SUBJECT: 20035
VARIABLE: 19
NUMBER OF SAMPLES:    12
VARIABLE SCALE:    1.00000
START POSITION:    62
STOP POSITION:    73
MEAN OF ERROR =       109.2500
ERROR DEVIATION =        84.4300
MEAN OF ERROR RATE =         0.0000
DEVIATION OF ERROR RATE =         8.2200
```

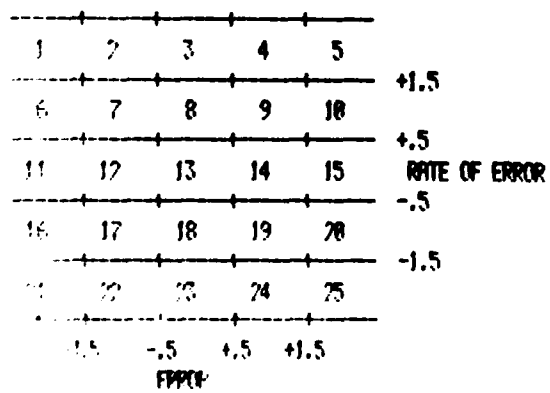
CELL VALUES



INDEX	ERROR	RATE OF ERROR	CELL
1	-39.250	12.000	6
2	-27.250	14.000	1
3	-13.250	16.000	2
4	2.750	9.000	7
5	11.750	0.000	12
6	11.750	-4.000	12
7	7.750	-4.000	12
8	3.750	-2.000	12
9	1.750	3.000	12
10	4.750	9.000	7
11	13.750	8.000	7

SUBJECT: 20035
VARIABLE: 19
NUMBER OF SAMPLES: 20
VARIABLE SCALE: 1.00000
START POSITION: 81
STOP POSITION: 100
MEAN OF ERROR = 37.8900
ERROR DEVIATION = 84.4300
MEAN OF ERROR RATE = 0.0000
DEVIATION OF ERROR RATE = 8.2200

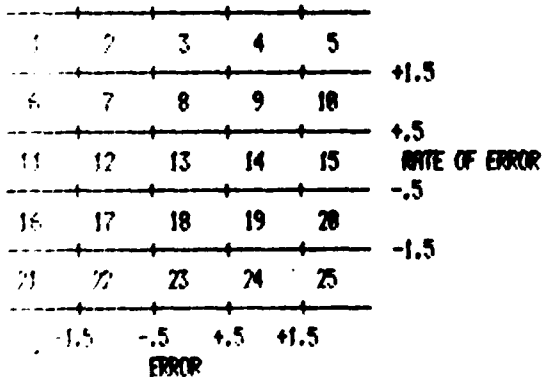
CELL VALUES



```
INDEX    ERROR  RATE OF ERROR  CELL

    1  145.110        -8.000   19
    2  137.110        -8.000   19
    3  129.110        -4.000   14
    4  125.110        -9.000   19
    5  116.110       -10.000   19
    6  106.110        -5.000   19
    7  101.110        -3.000   14
    8   98.110        -9.000   19
    9   89.110       -22.000   24
   10   67.110       -24.000   23
   11   43.110       -24.000   23
   12   19.110       -22.000   23
   13   -2.890       -21.000   23
   14  -23.890       -21.000   22
   15  -44.890       -16.000   22
   16  -60.890       -11.000   17
   17  -71.890       -12.000   17
   18  -83.890       -15.000   22
   19  -98.890       -14.000   21
   20 -112.890        -8.000   16
   21 -120.890         3.000   11
   22 -117.890         8.000    6
   23 -109.890        11.000    6
   24  -98.890        13.000    1
   25  -85.890        18.000    2
   26  -67.890        21.000    2
   27  -46.890        19.000    2
SUBJECT: 20005
PHASE: 19
NUMBER OF SAMPLES:    17
PHASE SCALE:    1.00000
START POSITION:   112
STOP POSITION:   123
MEAN OF ERROR =        86.4200
ERROR DEVIATION =       84.4300
MEAN OF ERROR RATE =        0.0000
DEVIATION OF ERROR RATE =        8.2200
```

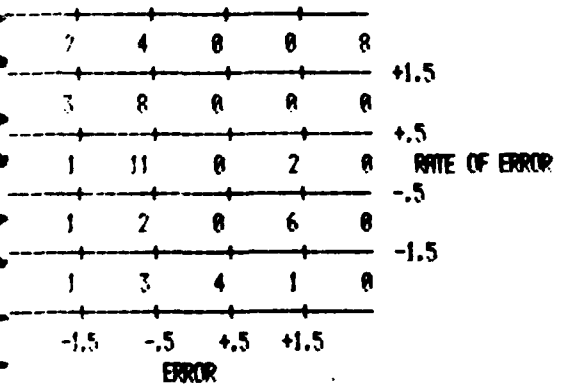
CELL VALUES



Example Output PHAN 3 (Continued)

+X	ERROR	RATE OF ERROR	CELL
1	-27.420	7.000	7
2	-20.420	5.000	7
3	-15.420	4.000	12
4	-11.420	4.000	12
5	-7.420	3.000	12
6	-4.420	5.000	7
7	0.580	7.000	7
8	7.580	9.000	7
9	16.580	3.000	12
10	19.580	1.000	12
11	20.580	1.000	12

CELL VALUES



CELL TRANSITION MATRIX:

0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	4	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	2	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	3	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	3	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.67	0.00	0.00	0.00	0.00	0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.67	0.00	0.00	0.00	0.00	0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.57	0.00	0.00	0.00	0.00	0.43	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.20	0.00	0.00	0.00	0.00	0.80	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.50	0.00	0.00	0.00	0.00	0.50	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33	0.00	0.00	0.00	0.00	0.50	0.00	0.00	0.00	0.00	0.17	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33	0.00	0.00	0.00	0.33	0.33	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.25	0.75	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00

Program Listing PHAN 4

```
FORTRAN IV     M01A-1   TUE 28-JUL-81 00:15:38          PAGE 001

                C     PHAN4.FOR
                C
                C     PURPOSE TO READ MULTIPLE COUNT MATRICES AND
                C     TO FORM ONE TRANSITION MATRIX
0001                  DIMENSION KTMATX(25,25),ITMATX(25,25),ATMATX(25,25)
0002                  DO 5 I=1,25
0003                  DO 6 J=1,25
0004                  KTMATX(I,J)=0
0005            6     CONTINUE
0006            5     CONTINUE
0007                  PAUSE'HIT RETURN & NAME OUTPUT FILE'
0008                  CALL ASSIGN(1,' ',-1,'NEW',' ',1)
0009           11     PAUSE'HIT RETURN & NAME INPUT FILE'
0010                  CALL ASSIGN(2,' ',-1,'OLD',' ',1)
0011                  DO 10 I=1,25
0012           10     READ(2,100)(ITMATX(I,J),J=1,25)
0013                  DO 12 I=1,25
0014                  DO 13 J=1,25
0015                  KTMATX(I,J)=KTMATX(I,J)+ITMATX(I,J)
0016           13     CONTINUE
0017           12     CONTINUE
                C     COMPUTE TRANSITION MATRIX
0018                  DO 14 I=1,25
0019                  SUM=0.
0020                  DO 15 J=1,25
0021                  ATMATX(I,J)=FLOAT(KTMATX(I,J))
0022           15     SUM=SUM+ATMATX(I,J)
0023                  DO 16 J=1,25
0024                  IF(SUM.EQ.0.) SUM=1.
0026           16     ATMATX(I,J)=ATMATX(I,J)/SUM
0027           14     CONTINUE
0028                  PAUSE'HIT RETURN &INPUT 1FOR CONTINUE,0 FOR STOP'
0029                  READ(5,101)IANS
0030                  CALL CLOSE(2)
0031                  IF(IANS.EQ.1) GO TO 11
0032                  DO 17 I=1,25
0034           17     WRITE(1,120)(ATMATX(I,J),J=1,25)
0035                  CALL CLOSE(1)
0036                  STOP
0037           100    FORMAT(2514)
0038           101    FORMAT(I1)
0039           120    FORMAT(25F5.2)
0040                  END
```

Table 7

PHAN 4.FOR

Purpose: Reads multiple count matrices and forms one transition matrix	
Computer Output	User Input
	• Run PHAN 4 ⟨CR⟩
Pause - Hit RETURN and name Output File	⟨CR⟩ *DK:H491T3.DAT ⟨CR⟩
Pause - Hit RETURN and name Input File	⟨CR⟩ *DK:H491C3.DAT ⟨CR⟩
Pause - Hit RETURN and Input 1 for Continue, 0 for Stop	⟨CR⟩ * 1 ⟨CR⟩
Pause - Hit RETURN and name Input File	⟨CR⟩ *DK:H201C3.DAT ⟨CR⟩
Pause - Hit RETURN and Input 1 for Continue, 0 for Stop	⟨CR⟩ * 0 ⟨CR⟩

```
FORTRAN IV     H01A-1    TUE 28-JUL-81 01:06:34          PAGE 001

      C     PHAN5.FOR
      C
      C     PURPOSE TO READ MULTIPLE COUNT MATRICES AND
      C     TO FORM ONE TRANSITION MATRIX
0001        DIMENSION KTMATX(25,25),ITMATX(25,25)
0002        DO 5 I=1,25
0003        DO 6 J=1,25
0004        KTMATX(I,J)=0
0005  6     CONTINUE
0006  5     CONTINUE
0007        PAUSE'HIT RETURN & NAME OUTPUT FILE'
0008        CALL ASSIGN(1,' ',-1,'NEW','NC',1)
0009  11    PAUSE'HIT RETURN & NAME INPUT FILE'
0010        CALL ASSIGN(2,' ',-1,'RDO','NC',1)
0011        DO 10 I=1,25
0012  10    READ(2,100)(ITMATX(I,J),J=1,25)
0013        DO 12 I=1,25
0014        DO 13 J=1,25
0015        KTMATX(I,J)=KTMATX(I,J)+ITMATX(I,J)
0016  13    CONTINUE
0017  12    CONTINUE
0018        PAUSE'HIT RETURN &INPUT 1FOR CONTINUE,0 FOR STOP'
0019        READ(5,101)IANS
0020        CALL CLOSE(2)
0021        IF(IANS.EQ.1) GO TO 11
0023        DO 17 I=1,25
0024  17    WRITE(1,100)(KTMATX(I,J),J=1,25)
0025        CALL CLOSE(1)
0026        STOP
0027  100   FORMAT(2514)
0028  101   FORMAT(I1)
0029        END
```

Table 8

PHAN 5.FOR

Purpose: Takes multiple count files and creates a single count file	
Computer Output	User Input
	• Run PHAN 5 ⟨CR⟩
Pause - Hit RETURN and name Output File	⟨CR⟩ *DK:HPC5.DAT ⟨CR⟩
Pause - Hit RETURN and name Input File	⟨CR⟩ *DK:H991C3.DAT ⟨CR⟩
Pause - Hit RETURN and Input 1 for Continue, 0 for Stop	⟨CR⟩ * 1 ⟨CR⟩
Pause - Hit RETURN and name Input File	⟨CR⟩ *DK:H201C3.DAT ⟨CR⟩
Pause - Hit RETURN and Input 1 for Continue, 0 for Stop	⟨CR⟩ * 0 ⟨CR⟩

Program Listing WTMAT

```
FORTRAN IV     M01A-1    TUE 28-JUL-81 08:09:51              PAGE 001

      C PROGRAM 'WTMAT.FOR'
      C CREATED FOR THE M11A SYSTEM
      C ORIGINAL DATE: 4-JAN-80
      C PURPOSE: STOCHASTIC ADJUSTMENT FOR THE 'MAP' SYSTEM
      C TRANSITION MATRICES AND THE SCORE ASSOCIATED WITH
      C EACH MATRIX ARE THE INPUT VARIABLES.
      C OUTPUT IS A WEIGHT MATRIX WHICH MAY BE USED TO SCORE
      C THE PERFORMANCE OF A SAMPLE.
0001        COMMON /RBLOCK/ TRANS,TSPROB,DLIMIT,
           1                DLHOLD,WEIGHT,
           2                PSCORE,CSCORE,GAIN,
           3                ADJINT,ERRADJ,ERRLIM,
           4                WLOW,WHIGH
0002        COMMON /IBLOCK/ IFRMT1,IFRMT2,IFRMT3,
           1                IDLFND,NUMMAT,ITMSIZ,
           2                MAXDLI,MAXSTT,IACCNE,
           3                MAINT,ISOTH,LENGTH,
           4                IBIG1,IBIG2,IBIG3,IPSCTI
0003        DIMENSION TRANS(2000),TSPROB(2000),
           1          DLIMIT(625),DLHOLD(625),
           2          WEIGHT(625),
           3          PSCORE(25),CSCORE(25),GAIN(25),IDLFND(25),
           4          IFRMT1(25),IFRMT2(25),IFRMT3(25)
0004        PAUSE 'INSERT DATA DISC IN UNIT ONE [1,4] AND HIT RETURN'
      C READ CONTROL VARIABLES
0005        CALL CUREAD
      C INITIALIZE WORK VARIABLES
0006        CALL INTIAL
      C READ THE PROBLEM VARIABLES
0007        CALL PUREAD
      C COMPUTE THE LIMITING DISTRIBUTIONS FOR
      C THE TRANSITION MATRICES
0008        CALL TMDLIM
      C COMPUTE THE TRANSTATE PROBABILITY MATRICES
0009        CALL TSPM
      C COMPUTE THE GAIN FACTOR FOR EACH MATRIX
0010        CALL TMGAIN
      C PERFORM THE STOCHASTIC ADJUSTMENT PROCEDURE
0011        CALL ADJUST
      C OUTPUT THE RESULTS
0012        CALL PWRITE
0013        STOP 'END OF PROGRAM STOADJ.FOR'
0014        END
```

Subroutine CVREAD

```
FORTRAN IV     M01A-1    TUE 05-MAY-81 03:50:57              PAGE 001

0001        SUBROUTINE CVREAD
      C PURPOSE IS TO READ THE CONTROL VARIABLES
0002        COMMON /RBLOCK/ TRANS,TSPROB,DLIMIT,
           1                DLHOLD,WEIGHT,
           2                PSCORE,CSCORE,GAIN,
           3                ADJINT,ERRADJ,ERRLIM,
           4                WLOW,WHIGH
0003        COMMON /IBLOCK/ IFRMT1,IFRMT2,IFRMT3,
           1                IDLFND,NUMMAT,ITMSIZ,
           2                MAXDLI,MAXRIT,IADONE,
           3                MAINT,ISOTM,LENGTH,
           4                IBIG1,IBIG2,IBIG3,IPSWT1
0004        DIMENSION TRANS(2000),TSPROB(2000),
           1         DLIMIT(400),DLHOLD(400),
           2         WEIGHT(400),
           3         PSCORE(20),CSCORE(20),GAIN(20),IDLFND(20),
           4         IFRMT1(20),IFRMT2(20),IFRMT3(20)
0005        WRITE(7,7100)
0006        CALL ASSIGN(1,' ',-1,'OLD','NC',1)
0007        WRITE(7,7110)
0008        CALL ASSIGN(2,' ',-1,'NEW','NC',1)
0009        WRITE(7,7120)
0010        READ(5,5100) NUMMAT
0011        WRITE(7,7130)
0012        READ(5,5100) ITMSIZ
0013        WRITE(7,7140)
0014        READ(5,5200) ADJINT
0015        WRITE(7,7150)
0016        READ(5,5200) ERRLIM
0017        WRITE(7,7160)
0018        READ(5,5200) ERRADJ
0019        WRITE(7,7170)
0020        READ(5,5100) MAXDLI
0021        WRITE(7,7180)
0022        READ(5,5100) MAXRIT
0023        WRITE(7,7190)
0024        READ(5,5300) (IFRMT1(I),I=1,20)
0025        WRITE(7,7200)
0026        READ(5,5300) (IFRMT2(I),I=1,20)
0027        WRITE(7,7210)
0028        READ(5,5300) (IFRMT3(I),I=1,20)
0029        WRITE(7,7220)
0030        READ(5,5200) WLOW
0031        WRITE(7,7230)
0032        READ(5,5200) WHIGH
0033        WRITE(7,7240)
0034        READ(5,5100) IPSWT1
0035        RETURN
0036  5100 FORMAT(I7)
0037  5200 FORMAT(F13.6)
0038  5300 FORMAT(20A2)
0039  7100 FORMAT(' NAME THE INPUT FILE'/)
0040  7110 FORMAT(' NAME THE RESULT FILE'/)
0041  7120 FORMAT(' ENTER THE NUMBER OF TRANSITION MATRICES (I)')
```

```
FORTRAN IV     H01A-1    TUE 05-MAY-81 03:58:57          PAGE 002

0042   7130 FORMAT(' ENTER SIZE OF ROW OR COLUMN FOR MATRICES (I)')
0043   7140 FORMAT(' ENTER INTIAL VALUE OF ENTRIES IN WEIGHT MATRIX (R)')
0044   7150 FORMAT(' ENTER ALLOWABLE ERROR IN LIMIT COMPUTATION (R)')
0045   7160 FORMAT(' ENTER ALLOWABLE ERROR IN SCORE DEVIATION (R)')
0046   7170 FORMAT(' ENTER MAXIMUM INTERATIONS FOR LIMIT COMPUTATION (I)')
0047   7180 FORMAT(' ENTER MAXIMUM INTERATIONS FOR WEIGHT ADJUSTMENT (I)')
0048   7190 FORMAT(' ENTER FORMAT FOR TRANSITION MATRICES READ (20A2)')
0049   7200 FORMAT(' ENTER FORMAT FOR SCORE READ (20A2)')
0050   7210 FORMAT(' ENTER FORMAT FOR WEIGHT OUTPUT (20A2)')
0051   7220 FORMAT(' ENTER LOWER LIMIT FOR WEIGHT MATRIX CELL (R)')
0052   7230 FORMAT(' ENTER HIGH LIMIT FOR WEIGHT MATRIX CELL (R)')
0053   7240 FORMAT(' ENTER INTERATION PRINT SWITCH FOR ADJUSTMENT (I)')
0054       END
```

```
FORTRAN IV     M81A-1    TUE 05-MAY-81 03:51:13          PAGE 001

0001          SUBROUTINE INTIAL
          C PURPOSE IS TO INTIALIZE VARIABLES IN COMMON
0002          COMMON /RBLOCK/ TRANS,TSPROB,DLIMIT,
             1               DLHOLD,WEIGHT,
             2               PSCORE,CSCORE,GAIN,
             3               ADJINT,ERRADJ,ERRLIM,
             4               WLOW,WHIGH
0003          COMMON /IBLOCK/ IFRMT1,IFRMT2,IFRMT3,
             1               IDLFND,NUMMAT,ITMSIZ,
             2               MAXDLI,MAXRIT,IADONE,
             3               MAINT,ISQTM,LENGTH,
             4               IBIG1,IBIG2,IBIG3,IPSWT1
0004          DIMENSION TRANS(2000),TSPROB(2000),
             1         DLIMIT(400),DLHOLD(400),
             2         WEIGHT(400),
             3         PSCORE(20),CSCORE(20),GAIN(20),IDLFND(20),
             4         IFRMT1(20),IFRMT2(20),IFRMT3(20)
0005          IBIG1=2000
0006          IBIG2=400
0007          IBIG3=20
0008          IADONE=0
0009          MAINT=0
0010          ISQTM=ITMSIZ*ITMSIZ
0011          LENGTH=NUMMAT*ITMSIZ*ITMSIZ
0012          DLINT=1.0/FLOAT(ITMSIZ)
0013          DO 100 I=1,IBIG1
0014          TRANS(I)=0.0
0015          TSPROB(I)=0.0
0016          IF(I.GT.IBIG2) GO TO 100
0018          WEIGHT(I)=ADJINT
0019          DLHOLD(I)=DLINT
0020          DLIMIT(I)=0.0
0021          IF(I.GT.IBIG3) GO TO 100
0023          PSCORE(I)=0.0
0024          CSCORE(I)=0.0
0025          GAIN(I)=0.0
0026          IDLFND(I)=0
0027      100 CONTINUE
0028          RETURN
0029          END
```

Subroutine PVREAD

```
FORTRAN IV     H01A-1    TUE 05-MAY-81 03:51:27          PAGE 001

0001          SUBROUTINE PUREAD
              C PURPOSE IS TO READ THE TRANSITION MATRICES AND THE
              C THE SCORE ASSOCIATED WITH EACH MATRIX
0002          COMMON /ABLOCK/ TRANS,TSPROB,DLIMIT,
             1               DLHOLD,WEIGHT,
             2               PSCORE,CSCORE,GAIN,
             3               ADJINT,ERRADJ,ERRLIM,
             4               WLOW,WHIGH
0003          COMMON /IBLOCK/ IFRMT1,IFRMT2,IFRMT3,
             1               IDLFND,NUMMAT,ITMSIZ,
             2               MAXDLI,MAXRIT,IADONE,
             3               MAINT,ISOTH,LENGTH,
             4               IBIG1,IBIG2,IBIG3,IPSWT1
0004          DIMENSION TRANS(2000),TSPROB(2000),
             1         DLIMIT(400),DLHOLD(400),
             2         WEIGHT(400),
             3         PSCORE(20),CSCORE(20),GAIN(20),IDLFND(20),
             4         IFRMT1(20),IFRMT2(20),IFRMT3(20)
0005          IK=NUMMAT*ITMSIZ
0006          I1=1
0007          DO 20 K=1,IK
0008          I2=I1+ITMSIZ-1
0009          READ(1,IFRMT1)(TRANS(I),I=I1,I2)
0010          I1=I1+ITMSIZ
0011    20    CONTINUE
0012          READ(1,IFRMT2) (PSCORE(I),I=1,NUMMAT)
0013          RETURN
0014          END
```

Subroutine TMDLIM

```
FORTRAN IV     M81A-1    TUE 05-MAY-81 03:51:40              PAGE 001

0001        SUBROUTINE TMDLIM
      C PURPOSE IS TO COMPUTE THE LIMITING DISTRIBUTION
      C FOR EACH OF THE TRANSITION MATRICES
0002        COMMON /RBLOCK/ TRANS,TSPROB,DLIMIT,
           1                DLHOLD,WEIGHT,
           2                PSCORE,CSCORE,GAIN,
           3                ADJINT,ERRADJ,ERRLIM,
           4                WLOW,WHIGH
0003        COMMON /IBLOCK/ IFRMT1,IFRMT2,IFRMT3,
           1                IDLFND,NUMMAT,ITMSIZ,
           2                MAXDLI,MAXRIT,IADONE,
           3                MAINT,ISQTM,LENGTH,
           4                IBIG1,IBIG2,IBIG3,IPSWT1
0004        DIMENSION TRANS(2000),TSPROB(2000),
           1          DLIMIT(400),DLHOLD(400),
           2          WEIGHT(400),
           3          PSCORE(20),CSCORE(20),GAIN(20),IDLFND(20),
           4          IFRMT1(20),IFRMT2(20),IFRMT3(20),
           5          TMHOLD(2000)
0005        DO 500 I=1,MAXDLI
0006        DO 300 J=1,NUMMAT
0007        IF(IDLFND(J).NE.0) GO TO 300
0009        KSTART=(J-1)*ITMSIZ+1
0010        KSTOP=KSTART+ITMSIZ-1
0011        SSQERR=0.0
0012        DO 200 K=KSTART,KSTOP
0013        LSTART=(K-1)*ITMSIZ+1
0014        LSTOP=LSTART+ITMSIZ-1
0015        DO 100 L=LSTART,LSTOP
0016        TMHOLD(L)=DLHOLD(K)*TRANS(L)
0017        WRITE(7,12) TMHOLD(L),DLHOLD(K),TRANS(L)
0018  12    FORMAT(3F10.3)
0019   100 CONTINUE
0020   200 CONTINUE
0021        KDUM=0
0022        DO 220 K=KSTART,KSTOP
0023        KDUM=KDUM+1
0024        LSTART=(J-1)*ISQTM+KDUM
0025        LSTOP=LSTART+ISQTM-1
0026        DLIMIT(K)=0.0
0027        DO 210 L=LSTART,LSTOP,ITMSIZ
0028        DLIMIT(K)=DLIMIT(K)+TMHOLD(L)
0029        WRITE(7,13)DLIMIT(K)
0030  13    FORMAT(F12.4)
0031   210 CONTINUE
0032        ERROR=DLIMIT(K)-DLHOLD(K)
0033        SSQERR=SSQERR+ERROR*ERROR
0034        DLHOLD(K)=DLIMIT(K)
0035   220 CONTINUE
0036        IF(SSQERR.LE.ERRLIM) IDLFND(J)=1
0038   300 CONTINUE
0039        IFSUM=0
0040        DO 400 M=1,NUMMAT
0041        IF(IDLFND(M).NE.0) IFSUM=IFSUM+1
```

Subroutine TMDLIM (Concluded)

```
FORTRAN IV     H01A-1    TUE 05-MAY-81 03:51:40          PAGE 002

0043    400 CONTINUE
0044        IF(IFSUM.EQ.NUMMAT) GO TO 800
0046    500 CONTINUE
0047        WRITE(7,7100)
0048        GO TO 999
0049    800 CONTINUE
0050        WRITE(7,7110) (I,IDLFND(I),I=1,NUMMAT)
0051        WRITE(7,7120)  (J,DLIMIT(J),J=1,ISUTM)
0052    999 CONTINUE
0053        RETURN
0054   7100 FORMAT(' FAILURE TO FIND ALL LIMITING DISTRIBUTIONS'/
           1       ' PROGRAM EXECUTION CONTINUES REGARDLESS')
0055   7110 FORMAT(' SUCCESSFUL COMPUTATION OF LIMITING DISTRIBUTIONS'/
           1       ' MATRIX',3X,'ITERATIONS'
           2       20(/1X,I5,9X,I5))
0056   7120 FORMAT(' VECTOR FOR LIMITS FOLLOWS:'/
           1       ' INDEX',6X,'LIMIT'
           1       200(/1X,I5,2X,F9.5))
0057        END
```

```
FORTRAN IV     M01A-1    TUE 05-MAY-81 03:51:57              PAGE 001

0001          SUBROUTINE TSPM
          C PURPOSE IS TO COMPUTE THE TRANSTATE PROBABILITY MATRICES
0002          COMMON /RBLOCK/ TRANS,TSPROB,DLIMIT,
             1                DLHOLD,WEIGHT,
             2                PSCOPE,CSCOPE,GAIN,
             3                ADJINT,ERRADJ,ERRLIM,
             4                WLOW,WHIGH
0003          COMMON /IBLOCK/ IFRMT1,IFRMT2,IFRMT3,
             1                IDLFND,NUMMAT,ITMSIZ,
             2                MAXDL1,MAXWIT,IADONE,
             3                NAINT,ISOTM,LENGTH,
             4                IBIG1,IBIG2,IBIG3,IPSWT1
0004          DIMENSION TRANS(2000),TSPROB(2000),
             1          DLIMIT(400),DLHOLD(400),
             2          WEIGHT(400),
             3          PSCOPE(20),CSCOPE(20),GAIN(20),IDLFND(20),
             4          IFRMT1(20),IFRMT2(20),IFRMT3(20),
             5          STPPRB(20)
0005          DO 300 I=1,NUMMAT
0006          STPPRB(I)=0.0
0007          JSTART=(I-1)*ITMSIZ+1
0008          JSTOP=JSTART+ITMSIZ-1
0009          DO 200 J=JSTART,JSTOP
0010          KSTART=(J-1)*ITMSIZ+1
0011          KSTOP=KSTART+ITMSIZ-1
0012          DO 100 K=KSTART,KSTOP
0013          TSPROB(K)=DLIMIT(J)*TRANS(K)
0014          STPPRB(I)=STPPRB(I)+TSPROB(K)
0015      100 CONTINUE
0016      200 CONTINUE
0017      300 CONTINUE
0018          WRITE(7,7100) (STPPRB(I),I=1,NUMMAT)
0019          WRITE(7,7110) (I,TSPROB(I),I=1,LENGTH)
0020          RETURN
0021     7100 FORMAT(' CHECK SUM FOR TRANSTATE MATRICES'/
             1       ' SHOULD BE EQUAL TO 1.0 (APPROXIMATELY)'/
             2       4(1X,5F10.5/))
0022     7110 FORMAT(' TRANSTATE VECTOR FOLLOWS:'/
             1       ' INDEX',9X,'VALUE',
             2       200(/1X,I5,1X,F13.5))
0023          END
```

Subroutine TMGAIN

```
FORTRAN IV     H01A-1    TUE 05-MAY-81 03:52:11                PAGE 001

0001        SUBROUTINE TMGAIN
      C PURPOSE IS TO COMPUTE THE GAIN FACTOR FOR EACH
      C OF THE TRANSITION MATRICES
0002        COMMON /RBLOCK/ TRANS,TSPROB,DLIMIT,
           1                DLHOLD,WEIGHT,
           2                PSCORE,CSCORE,GAIN,
           3                ADJINT,EPRADJ,EPRLIM,
           4                WLOW,WHIGH
0003        COMMON /IBLOCK/ IFRMT1,IFRMT2,IFRMT3,
           1                IOLFND,NUMMAT,ITMSIZ,
           2                MAXDLI,MAXRIT,IADONE,
           3                MAINT,ISQTM,LENGTH,
           4                IBIG1,IBIG2,IBIG3,IPSWT1
0004        DIMENSION TRANS(2000),TSPROB(2000),
           1          DLIMIT(400),DLHOLD(400),
           2          WEIGHT(400),
           3          PSCORE(20),CSCORE(20),GAIN(20),IOLFND(20),
           4          IFRMT1(20),IFRMT2(20),IFRMT3(20)
0005        DO 200 I=1,NUMMAT
0006        JSTART=(I-1)*ISQTM+1
0007        JSTOP=JSTART+ISQTM-1
0008        DO 100 J=JSTART,JSTOP
0009        GAIN(I)=GAIN(I)+TSPROB(J)*TSPROB(J)
0010    100 CONTINUE
0011    200 CONTINUE
0012        WRITE(7,7100) (GAIN(I),I=1,NUMMAT)
0013        RETURN
0014   7100 FORMAT(' COMPUTED GAIN FOLLOWS:'/
           1      4(1X,5F10.5))
0015        END
```

Subroutine ADJUST

```
FORTRAN IV     H01A-1    TUE 05-MAY-81 03:52:25          PAGE 001

0001          SUBROUTINE ADJUST
              C PURPOSE IS TO PERFORM THE WEIGHT MATRIX ADJUSTMENT
              C PROCEDURE UNTIL CONVERGANCE OR MAXIMUM
              C ITERATIONS ARE PERFORMED
0002          COMMON /RBLOCK/ TRANS,TSPROB,DLIMIT,
             1                DLHOLD,WEIGHT,
             2                PSCORE,CSCORE,GAIN,
             3                ADJINT,ERRADJ,ERRLIM,
             4                WLOW,WHIGH
0003          COMMON /IBLOCK/ IFRMT1,IFRMT2,IFRMT3,
             1                IDLFND,NUMMAT,ITMSIZ,
             2                MAXDLI,MAXAIT,IADONE,
             3                MAINT,ISQTM,LENGTH,
             4                IBIG1,IBIG2,IBIG3,IPSWT1
0004          DIMENSION TRANS(2000),TSPROB(2000),
             1          DLIMIT(400),DLHOLD(400),
             2          WEIGHT(400),
             3          PSCORE(20),CSCORE(20),GAIN(20),IDLFND(20),
             4          IFRMT1(20),IFRMT2(20),IFRMT3(20)
0005          SSEHLD=0.1E+29
0006          CLIMIT=0.1
0007          DO 500 I=1,MAXAIT
0008          MAINT=MAINT+1
0009          DO 300 J=1,NUMMAT
0010          CSCORE(J)=0.0
0011          DO 100 K=1,ISQTM
0012          L=(J-1)*ISQTM+K
0013          CSCORE(J)=CSCORE(J)+WEIGHT(K)*TSPROB(L)
0014      100 CONTINUE
0015          ERROR=PSCORE(J)-CSCORE(J)
0016          AJGAIN=ERROR/GAIN(J)
0017          DO 200 M=1,ISQTM
0018          N=(J-1)*ISQTM+M
0019          WEIGHT(M)=WEIGHT(M)+AJGAIN*TSPROB(N)
0020          IF(WEIGHT(M).LT.WLOW) WEIGHT(M)=WLOW
0022          IF(WEIGHT(M).GT.WHIGH) WEIGHT(M)=WHIGH
0024      200 CONTINUE
0025      300 CONTINUE
0026          SSQERR=0.0
0027          DO 320 J=1,NUMMAT
0028          CSCORE(J)=0.0
0029          DO 310 K=1,ISQTM
0030          L=(J-1)*ISQTM+K
0031          CSCORE(J)=CSCORE(J)+WEIGHT(K)*TSPROB(L)
0032      310 CONTINUE
0033          ERROR=PSCORE(J)-CSCORE(J)
0034          SSQERR=SSQERR+ERROR*ERROR
0035      320 CONTINUE
0036          IF(SSQERR.LE.ERRADJ) IADONE=1
0038          IF(IADONE.EQ.1) GO TO 330
0040          CINSSE=ABS(SSEHLD-SSQERR)
0041          IF(CINSSE.LT.CLIMIT) GO TO 330
0043          SSEHLD=SSQERR
0044          IF(MOD(MAINT,IPSWT1).NE.0) GO TO 500
```

```
 FORTRAN IV     H01A-1    TUE 05-MAY-81 03:52:25              PAGE 002

0046   330 CONTINUE
0047       WRITE(7,7100) MAINT,SSQERR
0048       DO 400 J1=1,NUMMAT
0049       WRITE(7,7105) J1,PSCORE(J1),CSCORE(J1)
0050   400 CONTINUE
0051       WRITE(7,7110)
0052       DO 410 K1=1,ISUBM
0053       WRITE(7,7115) K1,WEIGHT(K1)
0054   410 CONTINUE
0055       IF(IADONE.EQ.1) GO TO 600
0057       IF(CINSSE.LT.CLIMIT) GO TO 700
0059   500 CONTINUE
0060       WRITE(7,7120)
0061       GO TO 800
0062   600 CONTINUE
0063       WRITE(7,7130)
0064       GO TO 800
0065   700 CONTINUE
0066       WRITE(7,7140)
0067   800 CONTINUE
0068       RETURN
0069   7100 FORMAT(' ADJUST PASS: ',I4,' SUM OF SQUARED DEVIATION = ',F13.5/
          1       ' INDEX',8X,'SAMPLE',6X,'COMPUTED'/)
0070   7105 FORMAT(1X,I5,1X,F13.5,1X,F13.5)
0071   7110 FORMAT(' WEIGHT VECTOR FOLLOWS:'/
          1       ' INDEX',8X,'WEIGHT'/)
0072   7115 FORMAT(1X,I5,1X,F13.5)
0073   7120 FORMAT(' UNSUCCESSFUL ADJUSTMENT WITHIN CRITERIA',
          1       ' AND LIMIT ON ITERATIONS')
0074   7130 FORMAT(' ADJUSTMENT SUCCESSFULLY PERFORMED WITHIN CRITERIA')
0075   7140 FORMAT(' ADJUSTMENT TERMINATED DUE TO CONVERGENCE ',
          1       'TO MINIMUM POSSIBLE ERROR')
0076       END
```

Subroutine PWRITE

```
FORTRAN IV     H01A-1    TUE 05-MAY-81 03:52:43              PAGE 001

0001          SUBROUTINE PWRITE
          C PURPOSE IS TO WRITE THE RESULT FILE
0002          COMMON /RBLOCK/ TRANS,TSPPOB,DLIMIT,
             1                DLHOLD,WEIGHT,
             2                PSCORE,CSCORE,GAIN,
             3                ADJINT,ERRADJ,EPPLIM,
             4                WLOW,WHIGH
0003          COMMON /IBLOCK/ IFRMT1,IFRMT2,IFRMT3,
             1                IDLFND,NUMPAT,ITMSIZ,
             2                MAXDL1,MAXAIT,IADONE,
             3                NAINT,ISOTM,LENGTH,
             4                IBIG1,IBIG2,IBIG3
0004          DIMENSION TRANS(2000),TSPPOB(2000),
             1          DLIMIT(400),DLHOLD(400),
             2          WEIGHT(400),
             3          PSCORE(20),CSCORE(20),GAIN(20),IDLFND(20),
             4          IFRMT1(20),IFRMT2(20),IFRMT3(20)
0005          IK=ITMSIZ
0006          I1=1
0007          DO 20 K=1,IK
0008          I2=I1+ITMSIZ-1
0009          WRITE(2,IFRMT3)(WEIGHT(I),I=I1,I2)
0010          I1=I1+ITMSIZ
0011    20    CONTINUE
0012          RETURN
0013          END
```

Table 9

WTMAT. FOR

Stochastic Adjustment for the "Map" System	
Purpose: The computer takes the transition matrix and provides a score matrix.	
Computer Output	User Input
	• Run WTMAT ⟨CR⟩
Pause -- Insert data disc in unit one (DK:) and hit return	⟨CR⟩
Name the input file	*DK:TTEST1.DAT ⟨CR⟩
Name the result file	*DK:TTEST2.DAT ⟨CR⟩
Enter the number of transition matrices (I)[1]	2 ⟨CR⟩
Enter size of row or column for matrices (I)	5 ⟨CR⟩

[1] I = integer

Table 9

WTMAT . FOR (Continued)

Stochastic Adjustment for the "Map" System	
Computer Output	User Input
Enter initial value of entries in weight matrix (R)[2]	50. ⟨CR⟩
Enter allowable error in limit computation (R)	.001 ⟨CR⟩
Enter allowable error in score deviation (R)	5. ⟨CR⟩
Enter maximum interations for limit computation (I)	200 ⟨CR⟩
Enter maximum interations for weight adjustment (I)	200 ⟨CR⟩
Enter format for transition matrices read (20A2)	(5F3.1)[3] ⟨CR⟩

[2]R = real number. You must use a decimal point whenever the (R) appears.

[3]Whenever formats are indicated, parenthesis must be used.

Table 9

WTMAT. FOR (Concluded)

Stochastic Adjustment for the "Map" System	
Computer Output	User Input
Enter format for score read (20A2)	(2F5.1) 〈CR〉
Enter format for weight output (20A2)	(5F5.0) 〈CR〉
Enter lower limit for weight matrix cell (R)	0. 〈CR〉
Enter high limit for weight matrix cell (R)	99. 〈CR〉
Enter interation print switch for adjustment (I)	1 〈CR〉

Example Output WTMAT

WTMAT

Score Matrix

1	99.	99.	21.	99.
99.	50.	22.	99.	33.
70.	87.	67.	50.	50.
50.	99.	24.	99.	18.
9.9.	99.	61.	99.	99.

Program Listing MLREG1

```
FORTRAN IV     M01A-1    FRI 01-MAY-81 02:20:08              PAGE 001

      C PROGRAM 'MLREG1.FOR'
      C PURPOSE IS TO PERFORM A MULTIPLE LINEAR
      C REGRESSION ANALYSIS UTILIZING
      C SUBROUTINES FROM THE IMA SS PACKAGE
0001        DIMENSION X(6000),XBAR(20),STD(20),SCP(400),
           1          R(210),DSCP(20),RX(361),RY(19),IFORMT(20),
           2          RCOEF(19),SDRC(19),TVAL(19),ANS(10),
           3          WORK1(20),WORK2(20),ISAVE(20),IWORK1(19),IWORK2(19)
0002        DATA X/6000*0.0/,XBAR/20*0.0/,STD/20*0.0/,
           1     SCP/400*0.0/,IFORMT/20*' '/,R/210*0.0/,DSCP/20*0.0/,
           2     RX/361*0.0/,RY/19*0.0/,RCOEF/19*0.0/,
           3     SDRC/19*0.0/,TVAL/19*0.0/,ANS/10*0.0/,
           4     WORK1/20*0.0/,WORK2/20*0.0/,ISAVE/20*0/,
           5     IREAD/0/,IWORK1/19*0/,IWORK2/19*0
0003        PAUSE 'INSERT DATA DISC IN UNIT ONE (DY:) ## HIT RETURN'
0004        WRITE(7,7110)
0005        CALL ASSIGN(2,' ',-1,'RDO','NC',4)
0006        WRITE(7,7115)
0007        READ(5,5110) NOBS
0008        WRITE(7,7120)
0009        READ(5,5110) NVAR
0010        WRITE(7,7130)
0011        READ(5,5110) NIVAR
0012        WRITE(7,7140)
0013        READ(5,5120) (ISAVE(I),I=1,NIVAR)
0014        WRITE(7,7150)
0015        READ(5,5110) IDEPNT
0016        WRITE(7,7160)
0017        READ(5,5110) IUNIT
0018        WRITE(7,7170)
0019        READ(5,5130) (IFORMT(I),I=1,20)
0020        ISTOP=NOBS*(NVAR-1)+1
0021        DO 100 J=1,NOBS
0022        READ(2,IFORMT) (X(I),I=J,ISTOP,NOBS)
0023        IREAD=IREAD+1
0024        ISTOP=ISTOP+1
0025    100 CONTINUE
0026        CALL CORR(NOBS,NVAR,X,XBAR,STD,SCP,R,DSCP,WORK1,WORK2)
0027        CALL ORDER(NVAR,R,IDEPNT,NIVAR,ISAVE,RX,RY)
0028        CALL MINV(RX,NIVAR,DETRMT,IWORK1,IWORK2)
0029        CALL MULTR(NOBS,NIVAR,XBAR,STD,DSCP,RX,RY,ISAVE,
           1           RCOEF,SDRC,TVAL,ANS)
0030        CALL ORDER(NVAR,R,IDEPNT,NIVAR,ISAVE,RX,RY)
0031        CALL LWRITE(NOBS,NVAR,NIVAR,XBAR,STD,SCP,DSCP,RX,RY,ISAVE,RCOEF,
           1            SDRC,TVAL,ANS,IUNIT,IDEPNT)
0032        STOP 'END OF PROGRAM MLREG1.FOR'
0033   1000 FORMAT(32F8.3)
0034   5110 FORMAT(I7)
0035   5120 FORMAT(1914)
0036   5130 FORMAT(20A2)
0037   7110 FORMAT(' NAME THE INPUT FILE (ASSIGN)')
0038   7115 FORMAT(' ENTER NUMBER OF OBSERVATIONS (I)')
0039   7120 FORMAT(' ENTER NUMBER OF VARIABLES (I)')
0040   7130 FORMAT(' ENTER NUMBER OF INDEPENDENT VARIABLES (I)')
```

```
FORTRAN IV     H01A-1   FRI 01-MAY-81 02:20:38          PAGE 002

0041   7140 FORMAT(' ENTER INDICIES OF INDEPENDENT VARIABLES (1914)')
0042   7150 FORMAT(' ENTER INDEX OF DEPENDENT VARIABLE (I)')
0043   7160 FORMAT(' ENTER THE OUTPUT UNIT NUMBER (6=LP:,7=TT:)')
0044   7170 FORMAT(' ENTER THE INPUT FORMAT SPECIFICATION (20A2)')
0045        END
```

```
-FORTRAN IV     HP1A-1    FRI 01-MAY-81 02:20:41          PAGE 001

0001          SUBROUTINE CORR(N,M,X,XBAR,STD,RX,R,B,D,T)
0002          DIMENSION X(1),XBAR(1),STD(1),RX(1),R(1),B(1),D(1),T(1)
0003          DO 100 J=1,M
0004          B(J)=0.0
0005      100 T(J)=0.0
0006          K=(M*M+M)/2
0007          DO 102 I=1,K
0008      102 R(I)=0.0
0009          FN=N
0010          L=0
0011          DO 108 J=1,M
0012          DO 107 I=1,N
0013          L=L+1
0014      107 T(J)=T(J)+X(L)
0015          XBAR(J)=T(J)
0016      108 T(J)=T(J)/FN
0017          DO 115 I=1,N
0018          JK=0
0019          L=I-N
0020          DO 110 J=1,M
0021          L=L+N
0022          D(J)=X(L)-T(J)
0023      110 B(J)=B(J)+D(J)
0024          DO 115 J=1,M
0025          DO 115 K=1,J
0026          JK=JK+1
0027      115 R(JK)=R(JK)+D(J)*D(K)
0028          JK=0
0029          DO 210 J=1,M
0030          XBAR(J)=XBAR(J)/FN
0031          DO 210 K=1,J
0032          JK=JK+1
0033      210 R(JK)=R(JK)-B(J)*B(K)/FN
0034          JK=0
0035          DO 220 J=1,M
0036          JK=JK+J
0037      220 STD(J)=SQRT(ABS(R(JK)))
0038          DO 230 J=1,M
0039          DO 230 K=J,M
0040          JK=J+(K*K-K)/2
0041          L=M*(J-1)+K
0042          RX(L)=R(JK)
0043          L=M*(K-1)+J
0044          RX(L)=R(JK)
0045          IF(STD(J)*STD(K)) 225,222,225
0046      222 R(JK)=0.0
0047          GO TO 230
0048      225 R(JK)=R(JK)/(STD(J)*STD(K))
0049      230 CONTINUE
0050          FN=SQRT(FN-1.0)
0051          DO 240 J=1,M
0052      240 STD(J)=STD(J)/FN
0053          L=M
0054          DO 250 I=1,M
```

```
FORTRAN IV     H01A-1    FRI 01-MAY-81 02:20:44          PAGE 002

0055          L=L+M+1
0056      250 B(I)=RX(L)
0057          RETURN
0058          END
```

Subroutine ORDER

```
FORTRAN IV     H01A-1    FRI 01-MAY-81 02:20:59          PAGE 001

0001          SUBROUTINE ORDER(N,R,NDEP,K,ISAVE,RX,RY)
0002          DIMENSION R(1),ISAVE(1),RX(1),RY(1)
0003          MM=0
0004          DO 130 J=1,K
0005          L2=ISAVE(J)
0006          IF(NDEP-L2) 122,123,123
0007      122 L=NDEP+(L2*L2-L2)/2
0008          GO TO 125
0009      123 L=L2+(NDEP*NDEP-NDEP)/2
0010      125 RY(J)=R(L)
0011          DO 130 I=1,K
0012          L1=ISAVE(I)
0013          IF(L1-L2) 127,128,128
0014      127 L=L1+(L2*L2-L2)/2
0015          GO TO 129
0016      128 L=L2+(L1*L1-L1)/2
0017      129 MM=MM+1
0018      130 RX(MM)=R(L)
0019          ISAVE(K+1)=NDEP
0020          RETURN
0021          END
```

```
FORTRAN IV     H01A-1    FRI 01-MAY-81 02:21:12          PAGE 001

0001          SUBROUTINE MINV(A,N,D,L,M)
0002          DIMENSION A(1),L(1),M(1)
0003          D=1.0
0004          NK=-N
0005          DO 80 K=1,N
0006          NK=NK+N
0007          L(K)=K
0008          M(K)=K
0009          KK=NK+K
0010          BIGA=A(KK)
0011          DO 20 J=K,N
0012          IZ=N*(J-1)
0013          DO 20 I=K,N
0014          IJ=IZ+I
0015       10 IF(ABS(BIGA)-ABS(A(IJ))) 15,20,20
0016       15 BIGA=A(IJ)
0017          L(K)=I
0018          M(K)=J
0019       20 CONTINUE
0020          J=L(K)
0021          IF(J-K) 35,35,25
0022       25 KI=K-N
0023          DO 30 I=1,N
0024          KI=KI+N
0025          HOLD=-A(KI)
0026          JI=KI-K+J
0027          A(KI)=A(JI)
0028       30 A(JI)=HOLD
0029       35 I=M(K)
0030          IF(I-K) 45,45,38
0031       38 JP=N*(I-1)
0032          DO 40 J=1,N
0033          JK=NK+J
0034          JI=JP+J
0035          HOLD=-A(JK)
0036          A(JK)=A(JI)
0037       40 A(JI)=HOLD
0038       45 IF(BIGA) 48,46,48
0039       46 D=0.0
0040          RETURN
0041       48 DO 55 I=1,N
0042          IF(I-K) 50,55,50
0043       50 IK=NK+I
0044          A(IK)=A(IK)/(-BIGA)
0045       55 CONTINUE
0046          DO 65 I=1,N
0047          IK=NK+I
0048          IJ=I-N
0049          DO 65 J=1,N
0050          IJ=IJ+N
0051          IF(I-K) 60,65,60
0052       60 IF(J-K) 62,65,62
0053       62 KJ=IJ-I+K
0054          A(IJ)=A(IK)*A(KJ)+A(IJ)
```

```
. FORTRAN IV     H01A-1    FRI 01-MAY-81 02:21:12          PAGE 002

0055     65 CONTINUE
0056        KJ=K-N
0057        DO 75 J=1,N
0058        KJ=KJ+N
0059        IF(J-K) 70,75,70
0060     70 A(KJ)=A(KJ)/BIGA
0061     75 CONTINUE
0062        D=D*BIGA
0063        A(KK)=1.0/BIGA
0064     80 CONTINUE
0065        K=N
0066    100 K=K-1
0067        IF(K) 150,150,105
0068    105 I=L(K)
0069        IF(I-K) 120,120,108
0070    108 JQ=N*(K-1)
0071        JR=N*(I-1)
0072        DO 110 J=1,N
0073        JK=JQ+J
0074        HOLD=A(JK)
0075        JI=JR+J
0076        A(JK)=-A(JI)
0077    110 A(JI)=HOLD
0078    120 J=M(K)
0079        IF(J-K) 100,100,125
0080    125 KI=K-N
0081        DO 130 I=1,N
0082        KI=KI+N
0083        HOLD=A(KI)
0084        JI=KI-K+J
0085        A(KI)=-A(JI)
0086    130 A(JI)=HOLD
0087        GO TO 100
0088    150 CONTINUE
0089        RETURN
0090        END
```

```
FORTRAN IV     H01A-1    FPI 01-MAY-81 02:21:51          PAGE 001

0001        SUBROUTINE MULTR(N,K,XBAR,STD,D,RM,RV,ISAVE,B,SB,T,ANS)
0002        DIMENSION XBAR(1),STD(1),D(1),RM(1),RV(1),
           1          ISAVE(1),B(1),SB(1),T(1),ANS(1)
0003        MM=K+1
0004        DO 100 J=1,K
0005      100 B(J)=0.0
0006        DO 110 J=1,K
0007        LI=K*(J-1)
0008        DO 110 I=1,K
0009        L=LI+I
0010      110 B(J)=B(J)+RV(I)*RV(L)
0011        RM=0.0
0012        BO=0.0
0013        LI=ISAVE(MM)
0014        DO 120 I=1,K
0015        RM=RM+B(I)*RV(I)
0016        L=ISAVE(I)
0017        B(I)=B(I)*(STD(LI)/STD(L))
0018      120 BO=BO+B(I)*XBAR(L)
0019        BO=XBAR(LI)-BO
0020        SSAR=RM*D(LI)
0021        RM=SQRT(ABS(RM))
0022        SSDR=D(LI)-SSAR
0023        FN=N-K-1
0024        SV=SSDR/FN
0025        DO 130 J=1,K
0026        LI=K*(J-1)+J
0027        L=ISAVE(J)
0028      125 SB(J)=SQRT(ABS((RV(LI)/D(L))*SV))
0029      130 T(J)=B(J)/SB(J)
0030      135 SV=SQRT(ABS(SV))
0031        FK=K
0032        SSARM=SSAR/FK
0033        SSDRM=SSDR/FN
0034        F=SSARM/SSDRM
0035        ANS(1)=BO
0036        ANS(2)=RM
0037        ANS(3)=SV
0038        ANS(4)=SSAR
0039        ANS(5)=FK
0040        ANS(6)=SSARM
0041        ANS(7)=SSDR
0042        ANS(8)=FN
0043        ANS(9)=SSDRM
0044        ANS(10)=F
0045        RETURN
0046        END
```

```
FORTRAN IV      H01A-1    FRI 01-MAY-81 02:21:47                PAGE 03

      SUBROUTINE LWRITE(IREAD,NVAR,NIVAR,XBAR,STD,SCP,DSCP,RX,R,ISAVE,
     1                  RCOEF,SDRC,TVAL,ANS,IUNIT,INDEP)
      DIMENSION XBAR(1),STD(1),SCP(1),DSCP(1),R(1),RX(1),
     1          ISAVE(1),RCOEF(1),SDRC(1),TVAL(1),ANS(1)
      NVAR1=NIVAR+1
      WRITE(IUNIT,6110)
      WRITE(IUNIT,6120) IREAD
      WRITE(IUNIT,6130) NVAR
      WRITE(IUNIT,6140) NIVAR
      WRITE(IUNIT,6150) (ISAVE(I),I=1,NIVAR)
      WRITE(IUNIT,6160) ISAVE(NIVAR+1)
      WRITE(IUNIT,6170)
      DO 100 I=1,NIVAR
      J=ISAVE(I)
      WRITE(IUNIT,6180) J,XBAR(J),STD(J)
  100 CONTINUE
      WRITE(IUNIT,6190) XBAR(INDEP),STD(INDEP)
      WRITE(IUNIT,6191)
      DO 110 I=1,NVAR1
      DO 105 J=1,NVAR1
      I1=ISAVE(I)
      J1=ISAVE(J)
      IF(J1.LE.I1) GO TO 105
      K=(I1-1)*NVAR+J1
      WRITE(IUNIT,6192) I1,J1,SCP(K)
  105 CONTINUE
  110 CONTINUE
      WRITE(IUNIT,6193)
      DO 120 I=1,NIVAR
      DO 115 J=1,NIVAR
      IF(J.LE.I) GO TO 115
      K=(I-1)*NIVAR+J
      WRITE(IUNIT,6194) ISAVE(I),ISAVE(J),RX(K)
  115 CONTINUE
  120 CONTINUE
      WRITE(IUNIT,6195) ISAVE(NVAR1)
      DO 125 I=1,NIVAR
      WRITE(IUNIT,6196) ISAVE(I),R(I)
  125 CONTINUE
      WRITE(IUNIT,6197)
      DO 130 I=1,NVAR1
      J=ISAVE(I)
      IF(DSCP(J).GT.999999999.0) GO TO 129
      WRITE(IUNIT,6198) J,DSCP(J)
      GO TO 130
  129 CONTINUE
      WRITE(IUNIT,6199) J,DSCP(J)
  130 CONTINUE
      WRITE(IUNIT,6200)
      DO 200 I=1,NIVAR
      WRITE(IUNIT,6210) ISAVE(I),RCOEF(I),SDRC(I),TVAL(I)
  200 CONTINUE
      WRITE(IUNIT,6220) ANS(1)
      WRITE(IUNIT,6230) ANS(2)
```

```
FORTRAN IV     H01A-1    FRI 01-MAY-81 02:21:47            PAGE 002

0056        R2=ANS(2)**2
0057        WRITE(IUNIT,6231) R2
0058        WRITE(IUNIT,6240) ANS(3)
0059        WRITE(IUNIT,6250)
0060        WRITE(IUNIT,6260) ANS(4),ANS(5),ANS(6),ANS(10)
0061        WRITE(IUNIT,6270) ANS(7),ANS(8),ANS(9)
0062        RETURN
0063   6110 FORMAT(' '/' '/' MULTIPLE LINEAR REGRESSION'/)
0064   6120 FORMAT(' NUMBER OF OBSERVATIONS = ',I7)
0065   6130 FORMAT(' NUMBER OF VARIABLES = ',I7)
0066   6140 FORMAT(' NUMBER OF INDEPENDENT VARIABLES = ',I3)
0067   6150 FORMAT(' INDICES OF INDEPENDENT VARIABLES:'/
          1       ' ',1914)
0068   6160 FORMAT(' INDEX OF DEPENDENT VARIABLE: ',I4)
0069   6170 FORMAT(' '/' MEANS AND STANDARD DEVIATIONS OF INDEPENDENT '
          1          'VARIABLES:'/
          1       ' INDEX',8X,'MEAN',8X,'SDEV'/
          2       ' -----',8X,'----',8X,'----'/
          3       ' ')
0070   6180 FORMAT(1X,I5,1X,F11.3,1X,F11.3)
0071   6190 FORMAT(' '/' MEAN OF DEPENDENT VARIABLE = ',F11.3
          1          ' STANDARD DEVIATION OF DEPENDENT VARIABLE = ',F11.3)
0072   6191 FORMAT(' '/' SUM OF CROSS PRODUCTS:'
          1       ' INDEX',2X,'INDEX',7X,'SUMCP'
          2       ' -----',2X,'-----',7X,'-----'
          3       ' ')
0073   6192 FORMAT(1X,I5,2X,I5,1X,F11.3)
0074   6193 FORMAT(' '/' INDEPENDENT VARIABLE CORRELATIONS:'
          1       ' INDEX',2X,'INDEX',6X,'COEF'
          2       ' -----',2X,'-----',6X,'----'
          3       ' ')
0075   6194 FORMAT(1X,I5,2X,I5,1X,F9.3)
0076   6195 FORMAT(' '/' CORRELATIONS WITH DEPENDENT VARIABLE: ',I3
          1       ' INDEX',6X,'CORR'/
          2       ' -----',6X,'----'/
          3       ' ')
0077   6196 FORMAT(1X,I5,1X,F9.3)
0078   6197 FORMAT(' '/' SUM OF SQUARES:'/
          1       ' INDEX',9X,'SUMSQ'/
          2       ' -----',9X,'-----'/
          3       ' ')
0079   6198 FORMAT(1X,I5,1X,F13.3)
0080   6199 FORMAT(1X,I5,1X,F13.0)
0081   6200 FORMAT(' '/' REGRESSION COEFFICIENTS:'/
          1       ' INDEX',10X,'COEF',10X,'SDEV',5X,'T-VALUE'/
          2       ' -----',10X,'----',10X,'----',5X,'-------'/
          3       ' ')
0082   6210 FORMAT(1X,I5,1X,F13.7,1X,F13.7,1X,F11.3)
0083   6220 FORMAT(' '/' INTERCEPT = ',F11.3)
0084   6230 FORMAT(' '/' MULTIPLE CORRELATION COEFFICIENT = ',F6.3)
0085   6231 FORMAT(' '/' EXPLAINED VARIANCE = ',F6.3)
0086   6240 FORMAT(' '/' STANDARD ERROR OF ESTIMATE = ',F11.3)
0087   6250 FORMAT(' '/' ANALYSIS OF VARIANCE TABLE'/
          1       ' SOURCE',8X,'SUMSQ',5X,'DF',8X,'MEANSQ',6X,'F-RATIO')
```

```
FORTRAN IV      H01A-1    FRI 01-MAY-81 02:21:47              PAGE 003

          2       ' ------',8X,'-----',5X,'--',8X,'------',6X,'-------'
          3       ' ')
0088   6260 FORMAT(' REG',4X,F12.3,1X,F6.0,1X,F13.5,1X,F12.3)
0089   6270 FORMAT(' RES',4X,F12.3,1X,F6.0,1X,F13.5)
0090        END
```

Table 10

MLREG1.FOR

Multiple Linear Regression	
Purpose: Performs a multiple linear regression. Explanation: Enter # of observations = total # of observations Enter # of variables = total # of independent and dependent variables in your file Enter # of independent variables = total # you are going to use for this regression Indices of independent variables = the numbers that correspond to the independent variables in your file	
Computer Output	User Input
.	● Run MLREG1
Pause -- Insert data disc in unit one (DK:) and hit return	〈CR〉
Name the input file (Assign)	*DK:F20935.DAT 〈CR〉
Enter number of observations(I)	180 〈CR〉
Enter number of variables (I)	16 〈CR〉

. Table 10 ,

MLREG1.FOR (Concluded)

Multiple Linear Regression	
Computer Output	User Input
Enter number of independent variables (I)	3 〈CR〉
Enter indices of independent variables (1914)	8, 10, 15 〈CR〉
Enter index of dependent (I)	16 〈CR〉
Enter the output unit number (6 = LP, 7 = TT)	6 〈CR〉
Enter the input format specification (20A2)[1]	(16F20.10) 〈CR〉

[1]Whenever formats are indicated, parentheses must be used.

```
MULTIPLE LINEAR REGRESSION
NUMBER OF OBSERVATIONS =     174
NUMBER OF VARIABLES =        16
NUMBER OF INDEPENDENT VARIABLES =  2
INDICES OF INDEPENDENT VARIABLES:
  8  10
INDEX OF DEPENDENT VARIABLE:  16

MEANS AND STANDARD DEVIATIONS OF INDEPENDENT VARIABLES:
INDEX       MEAN        SDEV
-----       ----        ----

   8       0.287       0.292
  10       0.472       0.425

MEAN OF DEPENDENT VARIABLE =       0.406
STANDARD DEVIATION OF DEPENDENT VARIABLE =       0.381

SUM OF CROSS PRODUCTS:
INDEX  INDEX       SUMXP
-----  -----       -----

   8     10       17.489
   8     16       15.316
  10     16       18.960

INDEPENDENT VARIABLE CORRELATIONS:
INDEX  INDEX       CORR
-----  -----       ----

   8     10     0.582

CORRELATIONS WITH DEPENDENT VARIABLE:  16
INDEX      CORR
-----      ----

   8     0.796
  10     0.674

SUM OF SQUARES:
INDEX       SUMSQ
-----       -----

   8       14.777
  10       31.196
  16       25.000

REGRESSION COEFFICIENTS:
INDEX        COEF          SDEV      T-VALUE
-----        ----          ----      -------

   8     0.7941886     0.0670284      11.84
  10     0.2866171     0.0461322       6.21

INTERCEPT =       0.043

MULTIPLE CORRELATION COEFFICIENT =  0.837
```

EXPLAINED VARIANCE = 0.701

STANDARD ERROR OF ESTIMATE = 0.210

ANALYSIS OF VARIANCE TABLE

SOURCE	SUMSQ	DF	MEANSQ	F-RATIO
REG	17.569	2.	8.78448	199.983
RES	7.511	171.	0.04393	

```
      C PROGRAM 'HADMRD.FOR'
      C CREATED FOR THE V11A: 9-JAN-80
      C PURPOSE IS TO PERFORM A FAST HADAMARD TRANSFORM
      C FOR A VARIABLE NUMBER OF BOOLEAN TIME SEQUENCE RECORDS
      C CONTROL INPUT DEFINES THE SIZE OF THE HADAMARD MATRIX,
      C NUMBER OF BITS IN BOOLEAN TIME SEQUENCE, NUMBER OF
      C SHIFTS TO PERFORM ON BTS, AND THE LENGTH OF A SHIFT
      C THIS PROGRAM IS A PART OF THE ABSOLUTE COMPUTATION
      C PROCEDURE FOR THE 'MAP' SYSTEM
0001        COMMON /MBLOCK/ IB,IHDDIM,IPOW2
0002        DIMENSION IBTS(1024),IB(128),STAT(128,2)
0003        PAUSE 'INSERT DATA DISC IN UNIT ONE (DX:) AND HIT RETURN'
      C INPUT CONTROL VARIABLES
0004        WRITE(7,7100)
0005        CALL ASSIGN(1,' ',-1,'OLD','NC',1)
0006        WRITE(7,7110)
0007        CALL ASSIGN(2,' ',-1,'NEW','NC',1)
0008        WRITE(7,7120)
0009        READ(5,5100) IPOW2
0010        WRITE(7,7130)
0011        READ(5,5100) LENBTS
0012        WRITE(7,7140)
0013        READ(5,5100) NSHIFT
0014        WRITE(7,7150)
0015        READ(5,5100) LENSHF
      C COMPUTE OTHER CONTROL VARIABLES
0016        IHDDIM=2**IPOW2
0017        PSIZE=FLOAT(IHDDIM)
0018        NSHF1=NSHIFT+1
0019        PSSIZE=FLOAT(NSHF1)
0020        IREAD=0
      C READ ONE BTS RECORD
0021    100 CONTINUE
0022        READ(1,1100,END=700) (IBTS(I),I=1,LENBTS)
0023        IREAD=IREAD+1
0024        IF(LENBTS.GE.1024) GO TO 120
0026        ISTART=LENBTS+1
0027        DO 110 I=ISTART,1024
0028        IBTS(I)=0
0029    110 CONTINUE
0030    120 CONTINUE
      C COUNT NUMBER OF ONES AND ZEROS IN BOOLEAN TIME SEQUENCE
0031        IONE=0
0032        IZERO=0
0033        DO 130 I=1,LENBTS
0034        IF(IBTS(I).EQ.1) IONE=IONE+1
0035        IF(IBTS(I).EQ.0) IZERO=IZERO+1
0038    130 CONTINUE
      C INITIALIZE POPULATION TOTALS
0039        PHTOT=0.0
0040        PHTOT2=0.0
0041        PUTOT=0.0
0042        PUTOT2=0.0
      C INITIALIZE STATISTIC MATRIX
```

```
0043        DO 200 I=1,IHDDIM
0044        STAT(I,1)=0.0
0045        STAT(I,2)=0.0
0046    200 CONTINUE
        C SHIFT BTS TO WORKING VECTOR
0047        DO 500 I=1,NSHF1
0048        INDEX=0
0049        JSTART=(I-1)*LENSHF+1
0050        JSTOP=JSTART+IHDDIM-1
0051        DO 300 J=JSTART,JSTOP
0052        INDEX=INDEX+1
0053        IB(INDEX)=IBTS(J)
0054        IF(IB(INDEX).EQ.0) IB(INDEX)=(-1)
0056    300 CONTINUE
        C PERFORM FAST HADAMARD TRANSFORM
0057        CALL HADMRD
        C GATHER STATISTICAL DATA
0058        DO 400 J=1,IHDDIM
0059        STAT(J,1)=STAT(J,1)+FLOAT(IB(J))
0060        STAT(J,2)=STAT(J,2)+FLOAT(IB(J)**2)
0061    400 CONTINUE
0062    500 CONTINUE
        C OUTPUT THE SOLUTION FOR THE CURRENT BOOLEAN TIME SEQUENCE
0063        WRITE(7,7160) IREAD,LENBTS,IONE,IZERO,IHDDIM,NSHIFT,LENSHF,NSHF1
0064        WRITE(2,2100) IREAD,LENBTS,IONE,IZERO,IHDDIM,NSHIFT,LENSHF,NSHF1
0065        DO 600 I=1,IHDDIM
0066        SMEAN=STAT(I,1)/RSSIZE
0067        IF(NSHF1.LE.2) SVAR=0.0
0069        IF(NSHF1.GT.2) SVAR=(STAT(I,2)-(STAT(I,1)*STAT(I,1)/RSSIZE))
           1                    /(RSSIZE-1.0)
0071        WRITE(7,7170) I,STAT(I,1),STAT(I,2),SMEAN,SVAR
0072        WRITE(2,2110) I,SMEAN,SVAR
0073        PMTOT=PMTOT+SMEAN
0074        PMTOT2=PMTOT2+SMEAN*SMEAN
0075        PVTOT=PVTOT+SVAR
0076        PVTOT2=PVTOT2+SVAR*SVAR
0077    600 CONTINUE
0078        PMMEAN=PMTOT/PSIZE
0079        PMVAR=(PMTOT2-(PMTOT*PMTOT/PSIZE))/PSIZE
0080        PVMEAN=PVTOT/PSIZE
0081        PVVAR=(PVTOT2-(PVTOT*PVTOT/PSIZE))/PSIZE
0082        WRITE(7,7180) PMMEAN,PMVAR,PVMEAN,PVVAR
0083        WRITE(2,2120) PMMEAN,PMVAR,PVMEAN,PVVAR
        C READ NEXT BOOLEAN TIME SEQUENCE
0084        GO TO 100
        C END OF PROGRAM
0085    700 CONTINUE
0086        STOP 'END OF PROGRAM HADMRD.FOR'
0087  1100 FORMAT(12811)
0088  2100 FORMAT(8I5)
0089  2110 FORMAT(I3,2F9.3)
0090  2120 FORMAT(4F13.3)
0091  5100 FORMAT(I7)
0092  7100 FORMAT(' NAME THE INPUT FILE (ASSIGN)'/)
```

```
FORTRAN IV     H01A-1    FRI 08-MAY-81 00:39:48          PAGE 003

0093   7110 FORMAT(' NAME THE RESULT FILE (ASSIGN)'/)
0094   7120 FORMAT(' WHAT POWER OF 2 EQUALS HADAMARD SIZE (I)')
0095   7130 FORMAT(' WHAT IS LENGTH OF INPUT BOOLEAN TIME SEQUENCE (I)')
0096   7140 FORMAT(' HOW MANY SHIFTS ARE TO BE PERFORMED ON SAMPLE (I)')
0097   7150 FORMAT(' WHAT IS THE LENGTH OF A SHIFT (I)')
0098   7160 FORMAT(' FAST HADAMARD TRANSFORM'/
           1       ' INPUT BTS =       ',I7/
           2       ' LENGTH OF BTS =   ',I7/
           2       ' NUMBER OF ONES IN BTS = ',I7/
           2       ' NUMBER OF ZEROS IN BTS = ',I7/
           3       ' HADAMARD SIZE =   ',I7/
           4       ' NUMBER OF SHIFTS =',I7/
           5       ' LENGTH OF SHIFT = ',I7/
           6       ' SAMPLE SIZE =     ',I7/
           7       ' '/
           8       ' HADAMARD',7X,'SUMX',5X,'SUMXSQ',7X,'MEAN',3X,
           9       'VARIANCE'/
           9       ' -------',7X,'----',5X,'-------',7X,'----',3X,
           9       '--------'/
           9       ' ')
0099   7170 FORMAT(' ',1X,I7,2(1X,F10.0),2(2X,F9.3))
0100   7180 FORMAT(' POPULATION STATISTICS'/
           1       ' MEAN OF MEANS =          ',F13.3/
           2       ' VARIANCE OF MEANS =      ',F13.3/
           3       ' MEAN OF VARIANCES =      ',F13.3/
           4       ' VARIANCE OF VARIANCES = ',F13.3)
0101        END
```

```
0001        SUBROUTINE HADMRD
            C PURPOSE IS TO PERFORM A FAST HADAMARD TRANSFORM
            C ON AN INPUT BOOLEAN TIME SEQUENCE
0002        COMMON /HBLOCK/ IB,JSTOP,ISTOP
0003        DIMENSION IA(128),IB(128)
0004        DO 500 I=1,ISTOP
0005        INDEX=1
0006        JINC=2**I
0007        KSTOP=2**(I-1)
0008        DO 300 J=1,JSTOP,JINC
0009        DO 100 K=1,KSTOP
0010        LOC1=J+K-1
0011        LOC2=LOC1+KSTOP
0012        IA(INDEX)=IB(LOC1)+IB(LOC2)
0013        INDEX=INDEX+1
0014      100 CONTINUE
0015        DO 200 K=1,KSTOP
0016        LOC1=J+K-1
0017        LOC2=LOC1+KSTOP
0018        IA(INDEX)=IB(LOC1)-IB(LOC2)
0019        INDEX=INDEX+1
0020      200 CONTINUE
0021      300 CONTINUE
0022        DO 400 J=1,JSTOP
0023        IB(J)=IA(J)
0024      400 CONTINUE
0025      500 CONTINUE
0026        RETURN
0027        END
```

Program Listing RLOGIC

```
FORTRAN IV     H01A-1    FRI 08-MAY-81 00:29:49              PAGE 001

      C PROGRAM 'RLOGIC.FOR'
      C CREATED FOR THE H11A SYSTEM: 8-JAN-80
      C PURPOSE: ACCEPT A FILE OF BOOLEAN VARIABLES
      C AND A LOGICAL GOAL VARIABLE. COUNT THE NUMBER
      C OF OCCURENCES FOR ALL POSSIBLE COMBINATIONS
      C OF THE MINTERMS. COMPUTE THE PROBABILITIES
      C FOR THE EVENTS OF (1) MINTERM TRUE
      C (2) GOAL TRUE GIVEN MINTERM TRUE
      C (3) GOAL FALSE GIVEN MINTERM TRUE.
      C TEST FOR PURE LOGIC FUNCTION. WRITE OUTPUT
      C TO TERMINAL. WRITE OUTPUT FILE OF RESULTS.
      C THIS PROGRAM IS A PART OF THE RELATIVE
      C LOGIC PROCEDURE FOR THE 'MAP' SYSTEM.
0001        COMMON /BLOCK1/ PROB,KOUNT,KVALUE,PRBADJ,
     1                  IBSAMP,LOGIC,NVAR,IPURE,LENGTH,NSIZE
0002        DIMENSION PROB(1024,3),KOUNT(1024,3),KVALUE(1024),IBSAMP(10)
0003        PAUSE 'INSERT INPUT DATA DISC IN UNIT ONE (DK:) AND HIT RETURN'
      C NAME INPUT FILE
0004        WRITE(7,7100)
0005        CALL ASSIGN(1,' ',-1,'RDO','NC',1)
      C NAME OUTPUT FILE
0006        WRITE(7,7110)
0007        CALL ASSIGN(2,' ',-1,'NEW','NC',1)
      C ENTER NUMBER OF VARIABLES IN SAMPLE
0008        WRITE(7,7120)
0009        READ(5,5100) NVAR
      C ENTER THE COEFFICIENT ADJUSTMENT FACTOR
0010        WRITE(7,7130)
0011        READ(5,5110) PRBADJ
      C INITALIZE VARIBALES
0012        IPURE=0
0013        NSIZE=0
0014        LENGTH=2**NVAR
0015        DO 100 I=1,1024
0016        KOUNT(I,1)=0
0017        KOUNT(I,2)=0
0018        KOUNT(I,3)=0
0019        PROB(I,1)=0.0
0020        PROB(I,2)=0.0
0021        PROB(I,3)=0.0
0022        KVALUE(I)=-1
0023        IF(I.GT.10) GO TO 100
0025        IBSAMP(I)=0
0026    100 CONTINUE
      C READ ONE RECORD AND COUNT
0027    200 CONTINUE
0028        READ(1,1000,END=300) (IBSAMP(I),I=1,NVAR),LOGIC
0029        NSIZE=NSIZE+1
0030        CALL BTALLY
0031        GO TO 200
      C COMPUTE PROBABILTIES
0032    300 CONTINUE
0033        CALL RLPROB
      C TEST FOR PURITY OF LOGIC FUNCTION
```

Program Listing RLOGIC (Concluded)

```
FORTRAN IV     V01A-1    FRI 06-MAY-81 08:29:49          PAGE 002

0034        CALL TSTPUR
      C WRITE TO TERMINAL DEVICE
0035        CALL TWRITE
      C WRITE TO FILE DEVICE
0036        PAUSE 'INSERT OUTPUT DATA DISC IN UNIT ONE (DK:) AND HIT RETURN'
0037        CALL FWRITE
0038        STOP 'END OF PROGRAM RLOGIC.FOR'
0039   1000 FORMAT(1111)
0040   5100 FORMAT(I7)
0041   5110 FORMAT(F13.7)
0042   7100 FORMAT(' NAME THE INPUT FILE (ASSIGN)'/)
0043   7110 FORMAT(' NAME THE RESULT FILE (ASSIGN)'/)
0044   7120 FORMAT(' ENTER NUMBER OF VARIABLES FOR PROBLEM (1)')
0045   7130 FORMAT(' ENTER PROBABILITY FACTOR FOR ',
             1       'COEFFICIENT ADJUSTMENT (R)')
0046        END
```

Subroutine BTALLY

```
FORTRAN IV     H01A-1    FRI 08-MAY-81 08:30:04              PAGE 001

0001          SUBROUTINE BTALLY
              C PURPOSE IS TO COMPUTE LOCATION FOR ADDITION OF
              C CURRENT SAMPLE TO TOTAL AND THEN PERFORM ADDITION.
0002          COMMON /BLOCK1/ PROB,KOUNT,KVALUE,PRBADJ,
             1                IBSAMP,LOGIC,NVAR,IPURE,LENGTH,NSIZE
0003          DIMENSION PROB(1024,3),KOUNT(1024,3),KVALUE(1024),IBSAMP(10)
0004          INDEX=1
0005          DO 100 I=1,NVAR
0006          IF(IBSAMP(I).EQ.1) INDEX=INDEX+2**(I-1)
0008      100 CONTINUE
0009          KOUNT(INDEX,1)=KOUNT(INDEX,1)+1
0010          IF(LOGIC.EQ.1) KOUNT(INDEX,2)=KOUNT(INDEX,2)+1
0012          IF(LOGIC.NE.1) KOUNT(INDEX,3)=KOUNT(INDEX,3)+1
0014          RETURN
0015          END
```

Subroutine RLPROB

```
FORTRAN IV      H01A-1    FRI 03-MAY-81 08:30:16              PAGE 001

0001          SUBROUTINE RLPROB
      C PURPOSE IS TO COMPUTE THE PROBABILITIES FOR LOGIC FUNCTION
0002          COMMON /BLOCK1/ PROB,KOUNT,KVALUE,PRBAD,
             1                IBSAMP,LOGIC,NVAR,IPURE,LENGTH,NSIZE
0003          DIMENSION PROB(1024,3),KOUNT(1024,3),KVALUE(1024),IBSAMP(10)
0004          PSIZE=FLOAT(NSIZE)
0005          DO 100 I=1,LENGTH
0006          IF(KOUNT(I,1).LE.0) GO TO 100
0008          COUNT1=FLOAT(KOUNT(I,1))
0009          COUNT2=FLOAT(KOUNT(I,2))
0010          COUNT3=FLOAT(KOUNT(I,3))
0011          PROB(I,1)=COUNT1/PSIZE
0012          PROB(I,2)=COUNT2/COUNT1
0013          PROB(I,3)=COUNT3/COUNT1
0014      100 CONTINUE
0015          RETURN
0016          END
```

Subroutine TSTPUR

```
FORTRAN IV     H01A-1    FRI 08-MAY-81 00:30:29              PAGE 001

0001        SUBROUTINE TSTPUR
      C PURPOSE IS TO TEST FOR THE PURITY OF THE COMPUTED LOGIC FUNCTION
0002        COMMON /BLOCK1/ PROB,KOUNT,KVALUE,PRBADJ,
           1                IBSAMP,LOGIC,NVAR,IPURE,LENGTH,NSIZE
0003        DIMENSION PROB(1024,3),KOUNT(1024,3),KVALUE(1024),IBSAMP(10)
0004        DO 100 I=1,LENGTH
0005        IF(KOUNT(I,1).LE.0) KVALUE(I)=0
0007        IF(PROB(I,2).GE.PRBADJ) KVALUE(I)=1
0009        IF(PROB(I,3).GE.PRBADJ) KVALUE(I)=0
0011        IF(KVALUE(I).LT.0) GO TO 100
0013        IPURE=IPURE+1
0014    100 CONTINUE
0015        RETURN
0016        END
```

```
FORTRAN IV     H01A-1    FRI 08-MAY-81 00:30:41              PAGE 001

0001        SUBROUTINE TWRITE
      C PURPOSE IS TO WRITE TO TERMINAL
0002        COMMON /BLOCK1/ PROB,KOUNT,KVALUE,PRBADJ,
           1                IBSAMP,LOGIC,NVAR,IPURE,LENGTH,NSIZE
0003        DIMENSION PROB(1024,3),KOUNT(1024,3),KVALUE(1024),IBSAMP(10)
0004        WRITE(7,7100) NVAR,LENGTH,PRBADJ
0005        WRITE(7,7110) NSIZE,IPURE
0006        WRITE(7,7120)
0007        DO 100 I=1,LENGTH
0008        WRITE(7,7130) I,KOUNT(I,1),PROB(I,1),KVALUE(I),
           1              KOUNT(I,2),PROB(I,2),KOUNT(I,3),PROB(I,3)
0009    100 CONTINUE
0010        RETURN
0011    7100 FORMAT(' RELATIVE BOOLEAN LOGIC SEARCH FOR A LOGICIAL ',
           1        'FUNCTION OF THE MINTERMS'/
           2       ' NUMBER OF VARIABLES = ',I7/
           3       ' NUMBER OF MINTERMS =  ',I7/
           4       ' ADJUSTMENT FACTOR = ',F9.5)
0012    7110 FORMAT(' NUMBER OF SAMPLES =   ',I7/
           1       ' PURITY =              ',I7/
           2       ' PURITY SHOULD EQUAL THE NUMBER OF MINTERMS '/
           3       'FOR PURE LOGIC')
0013    7120 FORMAT(' ',2X,'MINTERM',4X,'TCOUNT',5X,'TPROB'/
           1          6X,'IDF',4X,'TCOUNT',
           2          5X,'TPROB',4X,'FCOUNT',5X,'FPROB'/
           3       ' ',2X,'-------',4X,'------',5X,'------',
           4          6X,'----',4X,'------',
           5          5X,'-----',4X,'------',5X,'------'/
           6     ' ')
0014    7130 FORMAT(' ',2X,I7,2X,I7,2X,F8.3,4X,I6,3X,I7,2X,F8.3,
           1          3X,I7,2X,F8.3)
0015        END
```

GLOSSARY OF TERMS

Subroutine FWRITE

```
FORTRAN IV     H01A-1    FRI 09-MAY-81 08:30:53          PAGE 001

0001          SUBROUTINE FWRITE
      C PURPOSE IS TO WRITE OUTPUT FILE
0002          COMMON /BLOCK1/ PROB,KOUNT,KVALUE,PRBADJ,
             1                IBSAPP,LOGIC,NUMR,IPURE,LENGTH,NSIZE
0003          DIMENSION PROB(1024,3),KOUNT(1024,3),KVALUE(1024),IBSAPP(10)
0004          WRITE(2,2100) NUMR,LENGTH,PRBADJ,NSIZE,IPURE
0005          DO 100 I=1,LENGTH
0006          WRITE(2,2110) I,KOUNT(I,1),PROB(I,1),KVALUE(I),
             1              KOUNT(I,2),PROB(I,2),KOUNT(I,3),PROB(I,3)
0007      100 CONTINUE
0008          RETURN
0009     2100 FORMAT(2I7,F13.7,2I7)
0010     2110 FORMAT(2I7,F8.3,I3,I7,F8.3,I7,F8.3)
0011          END
```

GLOSSARY OF TERMS

<CR>	Carriage Return
LP	Line Printer
TT	Teletype
(I)	Integer
(R)	Real number. A decimal point must be used with real numbers.
NTER ERMEAN	Enter Mean of Error
ERRDEV	Deviation of Error
RTMEAN	Mean Error Rate
RTDEV	Deviation Error Rate
FORMAT	Whenever a format is called for, parentheses must be used.