TRANSACTIONS, ILL. STATE ACAD. SCI.
Vol. 69, No. 3 (1976)

# BEHAVIORAL ITEM REDUCTION PROGRAM FOR PRECEDES/FOLLOWS CONTINGENCY TABLE FORMAT FOR DATAMYTE RECORDED BEHAVIORAL EVENTS

Jack A. Ward, Biological Sciences; Richard H. Miller[1], Psychology, Ramesh Chaudhari and Steve Skinner, Computer Services; Illinois State University, Normal 61761

## ABSTRACT

A unique program for computer reduction of behavioral data is described. The recording system is commercially available, inexpensive, portable, and requires no previous knowledge of electronics and engineering as previous data retrieval systems have required. The recorded behavioral data is simply interfaced with the computer and the program to be described reduces the data onto three N x N contingency tables which allows rapid interpretation of events recorded. The amount of information produced is greater than any other method currently available to ethologists and other behavioral scientists.

## INTRODUCTION

Observers of animal behavior are often limited by their inability to record several kinds of data simultaneously, e.g. frequency and duration. When this is possible the extraction of the information from charts or magnetic tape is a cumbersome and time consuming job. White (1971) made the initial break through in recording behavioral events on magnetic tape and subsequent interfacing to a computer. Dawkins (1971) gave instructions for a more portable system that also recorded behavioral events on magnetic tape and which was later played through a simpler interface to the computer. Both systems simultaneously recorded the time of behavioral events. Dawkins (1971) noted that his system could not record duration of behavioral events. Fernald and Heinecke (1974) described yet another system whereby the observer punched a key on an event recorder which was then directly punched on paper tape thereby eliminating the intervening step of recording first on magnetic tape. All of these earlier prototypes required considerable knowledge of engineering and electronics and the observer had to construct part of the hardware himself. These earlier systems did reduce the amount of time employed in collecting and tallying behavioral events, however, the computer printouts simply listed the sequence of behavior as it occurred when recorded. None of these systems actually reduced data into a form that allowed immediate interpretation of behavioral events.

---

[1] Current address: Department of Psychology, Washington University, St. Louis, Mo.

The system we describe is commercially available[2], inexpensive, portable, and requires no previous knowledge of electronics or enginnering to operate it. A new recording device[2] resembling a 10-key calculator permits accurate, simultaneous recording; depression of keys not only records digital information on magnetic tape, but simultaneously records the time of the event in relation to events that preceded or those to follow. Furthermore, we have designed a program that reduces the behavioral information recorded into a format that allows immediate interpretation of observations. The program first corrects data for internal and observer-generated errors and then reduces data on three N x N contingency tables. The first table is a precedes/follows frequency matrix that lends itself quickly to many methods of sequence analyses (Slater, 1973). The second contingency table scores time elapsed between all precedes/follows couplets and the third table is a time delayed matrix accumulating time between all possible couplets regardless of their precedes/follows relationship. The amount of information generated is greater than any other method currently available to ethologists and other behavioral scientists. Ward developed the concept of the program and its need in behavioral research; Miller (1975) developed the prototype for the correction portion of the program; while Ramesh Chaudhari and Steve Skinner designed and wrote the program.

## PROCEDURES

The program described below was constructed for use with the DataMyte DAK-8 portable data collector that interfaces the computer by the retrieval of magnetic tape recorded data via a coupler[2] onto paper punch tape. The reduction program is also suitable for use with the new DataMyte 9001 (solid-state) portable data collector that interfaces directly with the computer. The details given below for recording are necessary for smooth and uninterrupted operation of the reduction program and are tailored for the DataMyte DAK-8.

DATA COLLECTION (use of hardware):

IMPORTANT: Follow all directions.

A. DISENGAGE RECORDER FROM CHARGER ATTACHMENT -- Place a cassette tape into the recorder (tapes should always be bulk erased before use) -- set recorded in "PLAY" position for a few seconds until tape has tightened -- make sure, if the cassette is being used from the beginning that the tape

---

2) Manufactured by Electro/General Corporation, 14960 Minnetonka Industrial Road, Minnetonka, Minn. 55343

progresses beyond the clear leader to the dark recording surface. (Note: When using a cassette, whether at the beginning or at any point in its length, position the recorder in play position for a few seconds to tighten the tape.)

B. CONNECT KEYBOARD TO RECORDER -- When the recorder is connected to the keyboard it will not operate in "fast forward", "rewind", or "play" modes; to do so it must be disconnected from the keyboard.

C. CHECK TO SEE THAT CLOCK IS IN "ON" POSITION -- Turn keyboard over - lever should be in the direction of the arrow for the "on" position.

D. PUT RECORDER IN "RECORD" MODE

E. TURN POWER SWITCH ON KEYBOARD TO "ON" POSITION - WHEN "ON" RED LIGHTS WILL APPEAR IN DISPLAY WINDOW -- If red lights flash this indicates low voltage and any data recorded will be lost; recharge recorder a minimum of two hours or obtain a freshly charged battery pack.

F. INTRODUCTORY INFORMATION -- The program prior to data entry reads for introductory information fields with a maximum of 10 digits per field; you must enter information into all fields specified to initiate data acquisition. Any information may be entered into the nine fields that the observer wishes, however, the program PRINTS the following when the data are actually reduced:

Experimenter's # :

Description:

# of Variables:

# of Subjects:

Date and Time:

Machine Ser # :

Total Exp Time:

FIELD #

# 1 enter NUMBER OF VARIABLES - followed by ENTER (number of variables limited to 25!)

# 2 NUMBER OF SUBJECTS - ENTER

# 3* MONTH - ENTER e.g. 7

# 4* DAY OF MONTH - ENTER e.g. 4

# 5* YEAR - ENTER e.g. 1776

# 6* TIME OF DAY (hour, minute) - ENTER e.g. 1750 = 5:50 p.m.

# 7 SERIAL NUMBER - enter serial number found on back of keyboard - - ENTER

# 8 EXPERIMENTER NUMBER- ENTER

# 9 ANY OTHER DESCRIPTIVE DATA YOU WISH: not to exceed 6 digits---ENTER

* = fields 3 to 6 printed as one line "Date and Time"

FIELD #10 will be read as the first data entry. BUT FIRST --- TURN KEYBOARD POWER SWITCH "OFF"! When it is turned back on, either 2 seconds or 2 hours later, it will reset the clock to zero and is a crucial step. If the observer forgets this step the correction portion of the program will correct the time of the 10th field as 0000, and you will be warned of this error. Elapsed time during the observation will be accumulated by the recorder. If time duration is crucial the observer must use an external timer.

WHEN YOU ARE READY TO BEGIN YOUR EXPERIMENT - TURN POWER SWITCH TO "ON". Enter your first event. This entry will establish ZERO TIME (0000).

DATUM #

1 45 (do not exceed 10-digit codes; maximum number of data points per observation period = 500) - ENTER

2 12 - ENTER

'   '
'   '
'   '
'   '

500 Etc.- ENTER

When the LAST datum is entered, or experiment ends, the observer must signal "the end" of data by - - - - - - -

9999 - ENTER

TURN POWER "OFF" - This resets the clock and the observer can repeat all steps for a second observation period. The recorder only functions when actual data is entered. A 20-minute observation period may be represented by only 2 minutes of tape. Remove cassette without rewinding - if used again it will be at a point ready for continued recording. The tape can be rewound, but if it is to be used again without first being bulk erased, the exact location where data entry concluded before must be located. Play the tape and adjust the volume on the recorder so the digital tones can be heard of your previous recording. A silence of tones will be unused tape.

ERROR MESSAGE - You can correct information entered wrong. For example, suppose a 44 - ENTER, should have been a 4 - ENTER. It could be corrected in the following manner - - - - - - -

44 - ENTER, 0 - ENTER, 4 - ENTER

Thus, the <u>0 - ENTER</u> following a datum removes that datum and it will be replaced by the correction given after 0 - ENTER. The corrected datum will be designated as occurring at the same time at which the erroneous datum was entered.

The number of corrections that can be made is infinite as long as two error signals do not follow one another, for example, 0 - ENTER; 0 - ENTER.

(The correction of Introductory Information can be made in the same way. Again, the number of corrections possible is infinite.)

RECORD-PLAY BACK COMPATIBILITY - If the original DataMyte recording equipment is used it is recommended that the recorder used to record data is the same recorder used to interface the coupler. Alignment varies from one recorder to another and using a different recorder to interface the coupler may result in erroneous "nonsense" data. It is impossible to keep recorders perfectly aligned at all times. Alignment can also be a problem even when using the same recorder if the experimenter allows a considerable time lapse between data recording and transcription. Therefore, it is also recommended that data on cassette tapes be transcribed and committed to a more permanent file form (e.g. paper punch tape) as soon after recording data as possible.

<u>REDUCTION OF DATA</u> (use of software):

In most instances the data contained on the magnetic tape for each observation period will be transferred to punch paper tape. Whatever method is chosen to interface the computer it is recommended that temporary or permanent data files be created for each observation period. It is further assumed in what follows that data manipulation, reduction, and summation will be done at a remote terminal.

The reduction program constructs precedes/follows contingency tables and has two parts -- first the data file is corrected and then secondly the data is reduced and displayed on N x N contingency tables, the process is continuous, however.

When the program is called up at a remote terminal it addresses the operator with a number of possible options which are:

HOW MANY FILES DO YOU WANT TO ACCUMULATE?

WHAT IS THE MAXIMUM CODE? (Select a realistic maximum that is in agreement with your code system - if the codes were 1 through 25, for example, the maximum should be selected as 25. Erroneous code words will be counted as variables and added to the final matrices. Only 25 variables are allowable and real code words plus erroneous codes may exceed this maximum.)

WHAT IS THE MAXIMUM TIME? (Choose a realistic time maximum, one that corresponds to the real time limit of the observation periods.)

DO YOU WANT A DATA LIST?

WHAT IS THE NAME OF THE NEXT FILE? (This statement will be repeated an equal number of times to the number of data files you are accumulating.)

I. CORRECTION OF DATA

A number of errors can occur either by entry of erroneous data by the observer or in many cases "nonsense" data entry occurs as an idiosyncrasy of the recorder and recording system. The program corrects for nine sources of potential error. Most corrections are not corrections per se - the program simply eliminates the entire entry that has certain specified illegal parameters. If paper punch tape is used the observer has the option of making manual corrections in which case the erroneous entries can be transformed into legitimate entries, particularly if the observer can recall the missing datum and its inclusion is desirable. In our experience the amount of errors generated is most often less than .01 of the total amount of data entered. Whether the observer chooses to ask for a data list or not the following corrections will be made on the data file(s) called up. If a data list is asked for the location of the error and its kind is noted.

1. ILLEGAL CHARACTER OR BLANK OCCURRED: For example the skip key on the DataMyte recorder produces an "*" -- an illegal character.

2. CODE MISSING OCCURRED: Cases where time is not preceded by a code word.

3. TIME MISSING OCCURRED: Cases where a code word is not followed by time.

4. CODE LONGER THAN TEN DIGITS OCCURRED: Maximum code word size.

5. NON-SEQUENTIAL TIME OCCURRED: An idiosyncrasy of the recording system whereby erroneous time signals are recorded.

6. TIME GREATER THAN 9999 OCCURRED: The time limit of the internal clock in the DataMyte recorder.

7. INITIAL ENTRY MISSING OCCURRED: "Zero" time not specified in Field #10.

8. LAST CODE NOT 9999 OCCURRED:

9. CODE GREATER THAN MAXIMUM OCCURRED: Specified by the experimenter.

The above corrections remove all erroneous data entry that we have encountered using this system in the reduction of nearly 500 data files (observation periods). Unless it is desirable to retrieve a missing datum no manipulation of a data file is necessary prior to its introduction into the program routine.

## II. DATA REDUCTION

This portion of the program reduces the data onto three N x N contingency tables. The first table is a precedes/follows frequency matrix, the second scores time elapsed between all precedes/follows couplets, and the third is a time-delay matrix accumulating time between all possible couplets regardless of their precedes/follows relationship. The following sample will illustrate each matrix.

SAMPLE DATA (5 variables, code words 11 through 15)

12,0000

13,0009

14,0010

11,0013

15,0101

15,0115

15,0136

12,0150

FREQUENCY TABLE

| preceding event | | following event | | | | |
|---|---|---|---|---|---|---|
| * | 11 | 12 | 13 | 14 | 15 | SUM |
| 11 | | | | | 1 | 1 |
| 12 | | | 1 | | | 1 |
| 13 | | | | 1 | | 1 |
| 14 | 1 | | | | | 1 |
| 15 | | 1 | | | 2 | 3 |
| SUM | 1 | 1 | 1 | 1 | 3 | 7 |

The Frequency Table tallies events as they occurred during recording on a precedes/follows contingency table.

TIME TABLE 1

| * | 11 | 12 | 13 | 14 | 15 | SUM |
|---|---|---|---|---|---|---|
| 11 | | | | | 88 | 88 |
| 12 | | | 9 | | | 9 |
| 13 | | | | 1 | | 1 |
| 14 | 3 | | | | | 3 |
| 15 | | 14 | | | 35 | 49 |
| SUM | 3 | 14 | 9 | 1 | 123 | 150 |

In Time Table 1 the time elapsed between each precedes/follows event tallied on the previous Frequency Table is shown - useful for latency or time between behavioral events.

TIME TABLE 2

| * | 11 | 12 | 13 | 14 | 15 | SUM |
|---|---|---|---|---|---|---|
| 11 | | 137 | | | 88 | 225 |
| 12 | | | 9 | | | 9 |
| 13 | 4 | 141 | | 1 | 92 | 238 |
| 14 | 3 | 140 | | | 91 | 234 |
| 15 | | 14 | | | 35 | 49 |
| SUM | 7 | 432 | 9 | 1 | 306 | 755 |

In Time Table 2 the time elapsed between all possible permutations of couplets regardless of their precedes/follows relationship is shown. Time Table 2 matrix differs from Time Table 1 in many ways. For example, the couplet 12 following 13 was not a sequential event in the Frequency Table or Time Table 1, however, it is a possible permutation of the variables used and the time elapsed between the two was 141 secs. This matrix is useful for "on" and "off" events and in this instance 13 could signal the beginning of some event which might have consisted of events 14, 11, and 15, and 12 signalled the end ("off") of that particular sequence.

It is doubtful that any experimenter would be interested in every value in the three matrices. However, the information is tallied and available if the need arises. The Frequency Table lends itself quickly to the computation of transitional probabilities and other methods of sequence analysis (Slater, 1973). The column or row totals can be easily used in routine statistical analyses such as analysis of variance, regression analysis, etc. Matrices can be created for each observation period or several periods may be accumulated on one set of matrices.

The system does have two disadvantages, 1) two behavioral events cannot be recorded simultaneously, and 2) each "ENTER" command on the keyboard must be followed by a 0.5 sec. interval before another key is depressed. We feel the advantages of the system considerably outweigh the disadvantages.

The entire program is appended.

## POTENTIAL USERS

The program is permanently filed with the Mid-Illinois Computer Cooperative (MICC) at Edwardsville, Illinois. Institutions in Illinois that have MICC services may request permission to use the program by writing any one of the authors. The program is available on cassette tape and other formats directly from Electro/General Corporation, 14960 Minnetonka Industrial Road, Minnetonka, Minnesota 55343. Interested persons should write the company directly.

## ACKNOWLEDGMENTS

We would like to take this opportunity to acknowledge many who contributed in the development of this material. First, financial assistance for the actual design and writing of the program was supported by Ward's NSF grant GB-30338. Student helpers worked at terminals reducing actual behavioral data which tested various facets of the program and were paid by Ward's Illinois State University Research Grants 74-38 and 75-12. Dr. Kup Tcheng, Director of Academic and Research Consultation, Computer Services, is to be commended for his encouragement in this project and making his staff available for the design and writing of the program. We thank Ken Chow who assisted Richard H. Miller in writing the prototype for the correction portion of the program. The assistance of John F. King, Assistant Research Engineering-Psychology and the services of the Department of Psychology for use of their teletype terminals and their Data General NOVA 820 computer was indispensable in creating the prototype.

## REFERENCES

Dawkins, R. 1971. A cheap method of recording behavioral events, for direct computer-access. Behaviour 40:162-173.

Fernald, R. D. and P. Heinecke. 1974. A computer compatible multi-purpose event recorder. Behaviour 40:268-275.

Miller, R. H. 1975. An editor computer program for non-laboratory observations. Masters Thesis, Department of Psychology, Illinois State University, 57 pp.

Slater, P. J. B. 1973. Describing sequences of behavior. In: Perspectives in Ethology, Eds. P. P. G. Bateson and P. H. Klopfer. Plenum Press, New York, pp 131-153.

White, R. E. C. 1971. WRATS: A computer compatible system for automatically recording and transcribing behavioral data. Behaviour 40: 135-161.

```
*********************************COMMENTS*********************************
*
* DATA REDUCTION PROGRAM FOR THE BIOLOGICAL SCIENCES
*
* THIS PROGRAM EDITS DATA FILES FROM BIOLOGICAL
* EXPERIMENTS.  DATA INCLUDES INTRODUCTORY INFORMATION,
* BEHAVIOR CODE, AND TIME.  THE DATA IS TESTED FOR THE
* FOLLOWING ERRORS:
*      ERROR IN THE INTRODUCTORY INFORMATION
*      ILLEGAL CHARACTERS OR BLANKS
*      MISSING CODES
*      MISSING TIMES
*      CODES LONGER THAN TEN DIGITS
*      NON-SEQUENTIAL TIMES
*      TIMES GREATER THAN 9999
*      INITIAL ENTRIES MISSING
*      LAST CODE NOT 9999
*      CODES GREATER THAN MAXIMUM SPECIFIED BY THE USER
* CONSULT THE WRITE-UP FOR DETAILED INFORMATION
* CONCERNING THE ERROR CHECKS.
*
* ONCE EDITING IS COMPLETED, FREQUENCY TABLES AND TIME
* TABLES ARE PRINTED OUT.  MORE THAN ONE FILE MAY BE
* ACCUMULATED INTO THE TABLES.
*
* PROGRAM REQUESTED BY JACK WARD, PROFESSOR OF BIOLOGICAL
* SCIENCES, ILLINOIS STATE UNIVERSITY, NORMAL, ILL.
*
* PROGRAM DESIGNED AND WRITTEN BY RAMESH CHAUDHARI AND
* STEVE SKINNER, COMPUTER SERVICES, ILLINOIS STATE.
*
* NOTE:  CONTACT THE ABOVE NAMED INDIVIDUALS FOR
*        INFORMATION CONCERNING USING THE PROGRAM OR
*        PROGRAM MODIFICATION
*
*************************************************************************
PROGRAM RMS1(TAPE1,TAPE2,INPUT,OUTPUT,TAPE3)
INTEGER A,B,C,D
INTEGER COMMA,BLANK,ASTR
DIMENSION A(20),B(10),C(10),D(2)
DIMENSION IVAR(9)
DIMENSION IERROR(9)
DATA COMMA,BLANK,NINE,IZERO/1R,,1R ,1R9,1R0/
DATA NCD/99/, NTM/0000/
PRINT,*HOW MANY FILES DO YOU WANT TO ACCUMULATE*,
READ,NFL
PRINT,*WHAT IS THE MAXIMUM CODE*,
READ,MCD
PRINT,*WHAT IS THE MAXIMUM TIME*,
READ,MTM
PRINT,*DO YOU WANT A DATA LIST*,
READ,ILIST
IFL=0
1111 CONTINUE
```

```
DATA IFLG/0/
ISW=0
JSW=0
KNT=0
IFL=IFL+1
PRINT,*WHAT IS THE NAME OF THE NEXT FILE*,
READ, FLN
DO 198 I=1,9
198 IERROR(I)=0
CALL GET(5HTAPE1,FLN,0,0)
1 READ(1,100)A
IF(EOF,1)900,2
2 CONTINUE
IF(ILIST.EQ.3HYES)WRITE(3,1001)A
1001 FORMAT(1H ,20R1)
100 FORMAT(20R1)
DO 5 I=1,20
IS=A(I)
IF(IS.EQ.BLANK.OR.IS.EQ.COMMA.OR.(IS.GE.IZERO.AND.IS.LE.NINE))GO TO 5
IC=1
GO TO 40
5 CONTINUE
DO 10 I=1,10
IF(A(I).NE.COMMA)GO TO 10
I1=I
GO TO 15
10 CONTINUE
11 IC=1
GO TO 40
15 CONTINUE
IF(I1.LE.11)GO TO 16
IC=4
GO TO 40
16 I2=I1-1
IF(I2.LE.0)GO TO 21
DO 20 I=1,I2
IF(A(I).NE.BLANK)GO TO 25
20 CONTINUE
21 IC=2
GO TO 40
25 I3=I1+1
DO 30 I=I3,20
IF(A(I).EQ.COMMA)GO TO 35
IF(A(I).NE.BLANK)GO TO 45
30 CONTINUE
35 IC=3
40 CONTINUE
IERROR(IC)=IERROR(IC)+1
IF(ILIST.NE.3HYES)GO TO 1
101 FORMAT(1R ,10R1)
GO TO (41,42,43,44),IC
41 WRITE(3,800)
800 FORMAT(40H ---ILLEGAL CHARACTER OR BLANK RECORD---)
GO TO 1
```

```
42 WRITE(3,801)
801 FORMAT(19H ---CODE MISSING---)
GO TO 1
43 WRITE(3,802)
802 FORMAT(19H ---TIME MISSING---)
GO TO 1
44 WRITE(3,803)
803 FORMAT(33H ---CODE LONGER THAN 10 DIGITS---)
GO TO 1
45 CONTINUE
DO 47 I=1,10
B(I)=BLANK
47 C(I)=BLANK
K=10
DO 50 I=1,I2
J=I2-I+1
B(K)=A(J)
50 K=K-1
K=10
DO 55 I=I3,20
J=20-I+I3
IF(A(J).EQ.BLANK)GO TO 55
C(K)=A(J)
K=K-1
55 CONTINUE
ENCODE(20,102,D)B,C
102 FORMAT(20R1)
DECODE(20,103,D)ICD,ITM
103 FORMAT(2I10)
KNT=KNT+1
IF(ICD.EQ.9999)GO TO 998
IF(ICD.LE.MCD.OR.KNT.LT.10)GO TO 197
IF(ILIST.EQ.3HYES)WRITE(3,808)
808 FORMAT(38H---CODE IS GREATER THAN THE MAXIMUM---)
IERROR(9)=IERROR(9)+1
 GO TO 1
197 CONTINUE
IF(ITM.LE.MTM.OR.KNT.LE.10)GO TO 210
IF(ILIST.EQ.3HYES)WRITE(3,904)
904 FORMAT(26H ---NON-SEQUENTIAL TIME---)
IERROR(5)=IERROR(5)+1
GO TO 1
210 CONTINUE
IF (KNT.LE.1) GO TO 510
IF(ICD.NE.0)GO TO 200
ISW=1
GO TO 1
200 IF(ITM.NE.0)GO TO 202
JSW=1
GO TO 511
202 IF(JSW.NE.1)GO TO 500
IF(ITM-ITM1)211,211,212
211 IF(ILIST.EQ.3HYES)WRITE(3,804)
804 FORMAT(26H ---NON-SEQUENTIAL TIME---)
```

```
IERROR(5)=IERROR(5)+1
GO TO 1
212 IF(ITM-9999)400,400,215
215 IF(ILIST.EQ.3HYES)WRITE(3,805)
805 FORMAT(29H ---TIME GREATER THAN 9999---)
IERROR(6)=IERROR(6)+1
GO TO 1
400 CONTINUE
IF(ISW.EQ.1)GO TO 520
410 WRITE(2,)ICD,ITM
420 CONTINUE
ICD1=ICD
ITM1=ITM
GO TO 1
500 IF(KNT-10)510,511,511
510 IVAR(KNT)=ICD
GO TO 420
511 WRITE(2,) IVAR
JSW=1
IF(ITM.EQ.0)GO TO 410
WRITE(2,) NCD,NTM
IF(ILIST.EQ.3HYES)WRITE(3,806)
806 FORMAT(28H ---INITIAL ENTRY MISSING---)
IERROR(7)=IERROR(7)+1
GO TO 410
520 ITM=ITM1
ISW=0
GO TO 410
900 IF(ILIST.EQ.3HYES)WRITE(3,807)
807 FORMAT(28H ---LAST CODE IS NOT 9999---)
IERROR(8)=IERROR(8)+1
ICD=9999
998 WRITE(2,)ICD,ITM
CALL RETURN(5HTAPE1)
IF(IFL.EQ.NFL)IFLG=1
REWIND 2
WRITE(3,199)IERROR
199 FORMAT(/* ILLEGAL CHARACTER OR BLANK OCCURRED*,I13/,
+* CODE MISSING OCCURRED*,I27/* TIME MISSING OCCURRED*,I27/
+* CODE LONGER THAN TEN DIGITS OCCURRED*,I12/
+* NON-SEQUENTIAL TIME OCCURRED*,I20/
+* TIME GREATER THAN 9999 OCCURRED*,I17/
+* INITIAL ENTRY MISSING OCCURRED*,I18/
+* LAST CODE NOT 9999 OCCURRED*,I21/
+* CODE GREATER THAN THE MAXIMUM OCCURRED*,I10)
CALL RMS2(IFLG)
CALL RETURN(5HTAPE2)
IF(IFLG.EQ.0)GO TO 1111
STOP
END
SUBROUTINE RMS2(IFLG1)
DIMENSION IVC(26),IFR(26,26),ITMI(26,26),ITMII(26,26),IVT(26)
DIMENSION IFLG(25,25),INDX(26)
DATA IFLG/625*1/,INDX/1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,
```

```
+17,18,19,20,21,22,23,24,25,26/
DATA IVC/26*0/,IFR,ITMI,ITMII/2028*0/,IVT/26*9999/,N/26/
READ(2,)NV,NS,IMO,IDY,IYR,ITM,ISN,IEXP,IDES
WRITE(3,100)IEXP,IDES,NV,NS,IMO,IDY,IYR,ITM,ISN
100 FORMAT(*1EXPERIMENTER'S ? :*,I7/* DESCRIPTION :*,I12/
+* ? OF VARIABLES :*,I9/* ? OF SUBJECTS :*,I10/* DATE AND TIME :*,
+I2,1H/,I2,1H/,I4,2X,I4/* MACHINE SER ? :*,I10)
IVC(N)=9999999
READ(2,)I,I
 I=NCD+1
IF(I.GT.1)GO TO 11
READ(2,)IVC(1),IT
I=2
11 READ(2,)ICD,IT
IF(ICD.EQ.9999)GO TO 20
I1=I-1
DO 15 J=1,I1
IF(ICD.EQ.IVC(J))GO TO 19
15 CONTINUE
IVC(I)=ICD
I=I+1
19 IF(ENDFILE 1)20,11
20 NCD=I-1
J=J+1
WRITE(3,101)IT
101 FORMAT(* TOTAL EXP TIME  :*,I8,* SECONDS*)
REWIND 2
READ(2,)NV,NS,IMO,IDY,IYR,ITM,ISN,IEXP,IDES
READ(2,)IC,IT
IF(IT.NE.0)WRITE(3,)* INITIAL ENTRY MISSING*
IF(IC.EQ.9999)GO TO 99
READ(2,)IC1,IT1
CALL INDEX(IC1,IVC,NCD,I)
IVT(I)=IT1
111 READ(2,)IC,IT
IF(IC.EQ.9999) GO TO 131
CALL INDEX(IC,IVC,NCD,J)
IFR(I,J)=IFR(I,J)+1
ITMI(I,J)=ITMI(I,J)+IT-IT1
DO 120 K=1,NCD
IF(IVT(K).GE.IT.OR.IFLG(K,J).NE.1)GO TO 120
IFLG(K,J)=0
ITMII(K,J)=ITMII(K,J)+IT-IVT(K)
120 CONTINUE
DO 125 K=1,NCD
125 IFLG(J,K)=1
IVT(J)=IT
IT1=IT
I=J
GO TO 111
131 CONTINUE
IF(IFLG1.EQ.0)RETURN
DO  140 I=1,NCD
DO 140 J=1,NCD
```

```
IFR(I,N)=IFR(I,N)+IFR(I,J)
ITMI(I,N)=ITMI(I,N)+ITMI(I,J)
ITMII(I,N)=ITMII(I,N)+ITMII(I,J)
IFR(N,J)=IFR(N,J)+IFR(I,J)
ITMI(N,J)=ITMI(N,J)+ITMI(I,J)
ITMII(N,J)=ITMII(N,J)+ITMII(I,J)
140 CONTINUE
DO 145 I=1,NCD
IFR(N,N)=IFR(N,N)+IFR(I,N)
ITMI(N,N)=ITMI(N,N)+ITMI(I,N)
ITMII(N,N)=ITMII(N,N)+ITMII(I,N)
145 CONTINUE
146 J=0
DO 150 I=2,NCD
I1=I-1
IF(IVC(I1).LE.IVC(I))GO TO 150
ISAVE=IVC(I1)
IVC(I1)=IVC(I)
IVC(I)=ISAVE
ISAVE=INDX(I1)
INDX(I1)=INDX(I)
INDX(I)=ISAVE
J=J+1
150 CONTINUE
IF(J.GT.0)GO TO 146
CALL PRNTR(1,IFR,NCD,IVC,INDX,N)
CALL PRNTR(2,ITMI,NCD,IVC,INDX,N)
CALL PRNTR(3,ITMII,NCD,IVC,INDX,N)
RETURN
99 WRITE(3,)* NO DATA ON FILE*
RETURN
END
SUBROUTINE INDEX(IC,ICD,N,J)
DIMENSION ICD(26)
DO 10 I=1,N
IF(IC.NE.ICD(I)) GO TO 10
J=I
RETURN
10 CONTINUE
END
SUBROUTINE PRNTR(IRN,ICD,NV,IVC,INDX,N)
DIMENSION ICD(26,26),TTL(6),IVC(26),INDX(26),ITEMP(26,26)
DATA TTL/9HFREQUENCY,5HTABLE,10HTIME TABLE,2H 1,
+10HTIME TABLE,2H 2/
DATA BLNK/1H /,SUM/6H   SUM/,BTM/6H  ----/
DO 5 I=1,N
DO 5 J=1,N
5 ITEMP(I,J)=0
DO 10 J=1,N
K=INDX(J)
DO 10 I=1,N
10 ITEMP(I,J)=ICD(I,K)
DO 20 I=1,N
K=INDX(I)
```

```
DO 20 J=1,N
20 ICD(I,J)=ITEMP(K,J)
I=2*IRN-1
WRITE(3,100)TTL(I),TTL(I+1)
100 FORMAT(1H1,2A10)
NT=NV/9+1
J1=1
J2=0
DO 50 K=1,NT
IF(K.EQ.NT)GO TO 40
J2=J1+8
WRITE(3,101)(IVC(J),J=J1,J2)
101 FORMAT(//7H      *,9I6/)
DO 30 I=1,NV
30 WRITE(3,102)IVC(I),(ICD(I,J),J=J1,J2)
WRITE(3,104)(BTM,I=1,9)
WRITE(3,103)(ICD(N,J),J=J1,J2)
J1=J1+9
GO TO 50
40 J1=J2+1
J2=NV
WRITE(3,101)(IVC(I),I=J1,J2)
J=J2-J1+2
WRITE(3,105)(BLNK,I=1,J),SUM
105 FORMAT(1H+,29A6)
IF(J.GT.1)GO TO 43
DO 42 I=J12,J2
42 WRITE(3,102)IVC(I),ICD(I,N)
WRITE(3,104)(BTM,I=1,J)
WRITE(3,103)ICD(N,N)
GO TO 50
43 CONTINUE
DO 45 I=1,NV
45 WRITE(3,102)IVC(I),(ICD(I,J),J=J1,J2),ICD(I,N)
J=J2-J1+2
WRITE(3,104)(BTM,I=1,J)
104 FORMAT(1H ,6X,9A6)
WRITE(3,103)(ICD(N,J),J=J1,J2),ICD(N,N)
103 FORMAT(*    SUM*,9I6)
50 CONTINUE
102 FORMAT(1H ,10I6)
RETURN
END
READY.
```