NASA CR-
141666

THE UNIVERSITY OF TEXAS AT HOUSTON

M. D. ANDERSON HOSPITAL AND TUMOR INSTITUTE

DEPARTMENT OF BIOMATHEMATICS

A DATA STORAGE, RETRIEVAL

AND

ANALYSIS SYSTEM FOR ENDOCRINE RESEARCH

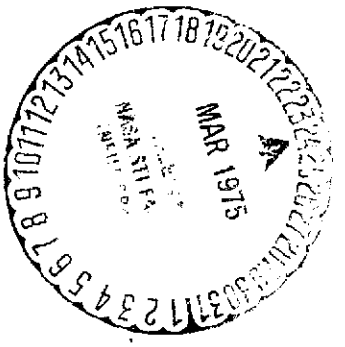
Prepared Under Contract NAS 9-13042

by

Lawrence E. Newton

and

Dennis A. Johnston, Ph.D.



for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

JOHNSON SPACE CENTER

ENDOCRINE LABORATORY

October, 1974

THE UNIVERSITY OF TEXAS AT HOUSTON

M. D. ANDERSON HOSPITAL AND TUMOR INSTITUTE

DEPARTMENT OF BIOMATHEMATICS

A DATA STORAGE, RETRIEVAL AND ANALYSIS SYSTEM FOR ENDOCRINE RESEARCH

Prepared Under Contract NAS 9-13042

by

Lawrence E. Newton

and

Dennis A. Johnston, Ph.D.

for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

JOHNSON SPACE CENTER

ENDOCRINE LABORATORY

October, 1974

THIS FORM MUST BE COMPLETED BY TYPEWRITER

01 4	01 7 PROGRAM NO.	JSC COMPUTER PROGRAM ABSTRACT	01 14 DATE (MMDDYY)
JSC	J263		11 27 74

01 20 TITLE OF PROGRAM (62 CHARACTERS MAXIMUM)	01 72 SYMBOLIC NAME (9 CHARACTERS MAXIMUM)	PARENT PROGRAM		
Storage, Retrieval and Analysis System for Endocrine Research		02 14 CATEGORY	02 16 SITE	02 19 PROGRAM NO

02 26 CATEGORY	02 27 LANGUAGE NO. 1	02 32 LANGUAGE NO. 2	02 37 KEY WORDS (8 MAXIMUM SEPARATED BY COMMAS)
J	FØR4		SKYLAB, ENDOCRINOLOGY, COMPUTER PROGRAM, INFORMATION RETRIEVAL, BIOASSAY, DATA MANAGEMENT, BIOMEDICAL DATA, BIOMETRICS

WHOM TO CONTACT ABOUT THE PROGRAM					05 48 STATUS	05 49
05 14 CONTACT (LAST NAME)	05 28 SITE	05 31 ORGN CODE	05 39 PROJECT NO	05 45 NASA CENTER	☐ A. UNDER DEVELOPMENT ☐ B. OPERATIONAL ☒ C. COMPLETED	☐ A. THIS PROGRAM IS NOT FOR SHARING ☐ B. LIMITED SHARING (SEE ABSTRACT)
Moseley	JSC	DB8	- -	JSC		

DATES		05 58 REVISION CODE	TIME AND COST FOR DEVELPOMENT			
05 50 INITIATED MMYY	05 54 COMPLETED MMYY		05 59 MAN-MONTHS	05 64 MACHINE HOURS	05 69 COMPUTER TYPE	05 74 TOTAL COST (DOLLARS)
07 72	11 74	☐ A. REVISION ☐ B. CANCELLATION	59 60 61 62 63	64 65 66 67 68	CDC 6000	2,0,0,0,0. 74 75 76 77 78 79 80

CARD NUMBER	COLUMN	ABSTRACT (ELITE MARGIN / PICA MARGIN)
06		This retrieval system builds, updates, retrieves, and
07		performs basic statistical analyses on blood, urine, and
08		diet parameters for the M071 and M073 Skylab and Apollo
09		experiments. This system permits data entry from cards
10		to build an indexed sequential file. Programs are easily
11		modified for specialized analyses. This system was developed
12		under Contract NAS 9-13042 by the Department of
13		Biomathematics, M. D. Anderson Hospital and Tumor Institute.
14		Principal Investigator - Dennis A. Johnston, Ph.D.
15		
16		
17		
18		
19		
20		
21		
22		
23		
24		
25		
26		
27		
28		
29		
30		
31		
32		
33		
34		
35		
36		
37		
38		
39		
40		
41		

RELATED DOCUMENTATION (66 CHARACTERS MAXIMUM SEPARATE EACH REF BY COMMAS)

42	

TABLE OF CONTENTS

1. INTRODUCTION

One major purpose of the Skylab program was to determine man's adaptability to prolonged spaceflight. NASA experiment M-073 was designed to consider man's response in the areas:

a. Fluid/electrolyte balance;

b. Regulation of calcium metabolism;

c. Regulation of metabolic processes; and a

d. General endocrine/metabolic adaption to a prolonged space environment.

A more detailed description of these studies may be found in Leach [1].

Contract NAS 9-13042 was negotiated to assist the NASA Endocrine Laboratory in the analysis of M-073. It was apparent from the outset of the contract that a storage and retrieval system would be necessary to maintain and statistically analyse the large amounts of diverse data which needed to be analysed. The original system was developed on a Xerox Data Systems Sigma 5 [2] and permitted simplified updating procedures and basic statistical analyses and graphs by astronaut of any selected parameter available on the system.

When M. D. Anderson converted from the XDS Sigma 5 to a Control Data Cyber 73, the system was converted also. This system

is documented in this report. The CDC version offers improved file handling abilities, improved throughput due to the increased speed and disk capabilities, and an ease of program change here-to-fore unavailable. This permits rapid and comprehensive analysis of all or selected parameters using general purpose programs with few modifications required. The method of modification is given in Section 7.

2. INPUT DATA FORMATS

All data input into the system is made via punched cards. Seventeen data forms have been developed for this purpose. Copies of these forms are listed in Appendix A. Form 0 is used to enter acceptable sample numbers, type of sample, and record time of collection. Forms 1, 4, 6, 7, 8, 9, 10, 11, and 12 are for parameters obtained from urine type samples. Forms 2, 3, and 5 are used for blood samples. Form 13 is used to record daily weight and water consumption; and Form 15 is used to record daily diet information. Form 14 is used for all updates and changes to all other forms except Form 13 which is updated using Form 14A. It should be noted that sample numbers refer to one and only one sample of either urine or blood (but not both) and that only the appropriate forms and update form type can be filled out using that sample number.

Decimal points are not required or recommended on the forms. Fields are integer unless an implied decimal point is indicated between card columns. For example, an entry of 0120 for EPI (epinephrine) on Form 1 would indicate a value of 12.0 μgm per total volume of sampled urine. Observe then that leading zeros are not required but that trailing zeros are preferred to quarantee the correct positioning of the number.

In this system, no zero, negative values, or totally blank data is valid. Instead, if data is not available or less than determin-

able, then one of the following codes are used:

-1 -- Test not performed

-2 -- Value is less than determinable

-3 -- Value to follow on update

-4 -- Calculated as combined values

-5 -- Less than a 24 hour period.

These entries should be justified as if they were data values. For example, use -020 for a less than determinable on EPI on Form 1. These are an indication to the computer system, as well as the user, as to why the datum is not present. A value of -2 will be interpreted by the computer as a test value of 0. Any other code will be printed on output but not included as an actual test value.

3. BUILDING AN INITIAL DATA FILE

Program Build is used to create an initial set of data files. Card number 76 in this program controls the function of the program. This card must set INITIAL=0 to create new files. Since this program then expects to create new files on mass storage the user must insure that none of the files to be built exist. See Section 3.1 for a sample deck.

There are four files required in the retrieval system and it is necessary that each of the files contain at least one record. File RA1 is the sample directory file and is a random file with the sample number as the key for each record. There must be one and only one record in this file for every sample. This file is built from sample directory cards which are punched from form zero. The sample directory cards must be in order of increasing sample number. For each sample number in file RA1 there must be a record with a corresponding sample number in either RA2, the Urine file, or RA3, the Blood file, but a given sample number must not appear in both RA2 and RA3. Both RA2 and RA3 are keyed on sample number.

Within files RA2 and RA3, each record is composed of data from one or more data cards. During creation of a file, it is not possible to replace an existing record, therefore, only one type of card for each file is permissable. It is suggested that only card forms 0, 1, 2 and 13 appear in a creation deck and that at least one of each of

these appear. The simplist way is to include only one of each card type in the initial creation, however, this will not produce an efficient file structure. Refer to section 2 for form numbers.

The build program will make few checks on the cards, but will print an error message if addition of an existing sample number is attempted. The program TYPLOK described later will perform further file integrity checks.

The fourth file, RA4, is the daily parameter file. This file is not associated with the sample directory. The key for each record in the file is the study number concatenated with the Julian date from the daily parameter card. The cards for this file (13 cards only) must be in ascending study order and ascending date within study.

A card with 99 in columns 1 and 2 or an end of file terminates the program.

3.1 Sample deck to build a new file (CREATE deck)

```
NASA,CM600000.
ACCOUNT,AN12318.
GET,OPL=BUILDPL.
MODIFY,LO=CET,F.
FTN,I.
LGO.
SAVE,RA1.
SAVE,RA2.
SAVE,RA3.
SAVE,RA4.
PACK,ZZZZZEF.
COPYSBF,ZZZZZEF,OUTPUT
EXIT.
PACK,ZZZZZEF.
COPYSBF,ZZZZZEF,OUTPUT.
7/8/9   (EOR)
*IDENT UPDT
*DECK BUILD
*D 76
7/8/9   (EOR)
        (Deck of data cards punched from forms
         in the order described in Section 3)
6/7/8/9 (EOI)
```

4. UPDATING AN EXISTING FILE

Program BUILD is used to update existing files. Source card number 76 must be set INITIAL=1 to update existing files. Any valid card in any order may be used to update a file.

Sample directory or daily parameter records, card codes 00 and 13, may be added, but existing records of this type cannot be replaced unless previously deleted. Sample directory records may be updated by use of 14 cards, and a daily parameter record may be updated by use of 14A cards.

Any complete record may be deleted by use of a 98 card. This will delete a record by file number and key.

All other record types may be added, or all of the information within a record which is contained on one card may be replaced by entering the card, or any field may be changed by use of a 14 card. The retrieval index shows which values within a record are contained on a particular card form and the value index to be used on a 14 card.

4.1 Checking Files

The program BUILD does not cross check the urine, blood and sample directory files. Program TYPELOK performs a number of cross

checks on these files and prints some tables showing the actual number of data points within the files. The output of this program is necessarily condensed and therefore intended as a programmers maintenence tool only.

4.2 Sample deck to modify an existing file (UPDATE deck)

```
NASA,CM60000.
ACCOUNT,AN12318.
GET,OPL=BUILDPL.
MODIFY,LO=GET,F.
FTN,I,L=0.
GET,RA1.
GET,RA2.
GET,RA3.
GET,RA4.
LGO.
REPLACE,RA1.
REPLACE,RA2.
REPLACE,RA3.
REPLACE,RA4.
PACK,ZZZZZEF.
COPYSBF,ZZZZZEF,OUTPUT.
EXIT.
PACK,ZZZZZEF.
COPYSBF,ZZZZZEF,OUTPUT.
7/8/9
*IDENT UPDT
*DECK BUILD
*D 77
7/8/9
          (Deck of cards for additions or
           modifications)
6/7/8/9
```

5. HOW TO RETRIEVE AND PROCESS THE DATA

The program is logically in two parts, the first part retrieves the data, the second part performs the analysis on the data. Data is retrieved for one mission (study) over one or more ranges of consecutive dates and for one test. Data is retrieved for man 1, 2, and 3 in order, and data for each individual is passed to the second part of the program one man at a time. Those statistical analyses which require data for all men simultaneously must modify a section of the statistics routine to save the necessary data. See section 7.

A basic series of statistical tests are performed for each man and varies depending on the available data. Data is sectioned into preflight, inflight and post flight time periods. All analyses assume date is the independent variable and test value the dependent variable.

The program requires a series of data cards to control the data retrieval and all data is retrieved from the four files described in sections 3 and 4. Appendix E shows the control card functions.

To retrieve data it is necessary to select the mission (study), the test type, the dates and the test name. By reference to Table E.3, the codes for the 5 available studies and the 9 test types may be found. Reference to Table B1 will show the code number index for the various tests. It is necessary to know which type of sample was used for a given test. For instance, to obtain the information on Blood insulin, test number 304, it is necessary to realize that only blood sample types are meaningful. Blood sample types are codes 5, 6, 7 or 9 from Table E.3. It is possible to retrieve on all 4 possibilities and the program will generate a separate report for each type indicating the empty types. Program TYPLOK will produce a table of the number of entries by sample, type and study. The TYPLOK printout currently shows that for Skylab 3 (study 3) and Blood insulin (test 304) that only plasma samples (type 5) were evaluated. It is then necessary to know the Julian dates of interest. Table E.3 shows the inclusive Julian dates of the various studies. By refering to these tables and the control card descriptions the following cards will retrieve the desired information.

JOB DATE 1300	Job card requesting date vs data Run number 1300 greater than 1000 indicates plots
STUDY EACH 3	Study 3 (Skylab 3)
DATE ALL	All days
TYPE EACH 5	Only type 5 (Plasma) samples are available
TEST EACH 304	Test 304 (Insulin)
END	End of Job
EOF	End of Control Cards.

6. DESCRIPTION OF THE BASIC STATISTICAL ANALYSIS PROGRAM

Several basic statistical tests, contrasts, and comparisons are made routinely when the statistical analysis of a test is requested. These analyses are performed on each astronaut separately. Appendix C contains a sample analysis for urinary calcium for the Commander of Skylab III. The analyses are performed on the Commander, Pilot and Support Pilot (or support person if requested) in order and consist of:

1. A diagnostic heading which contains:
   a. Retrieval parameters LSTUDY (number of study selected), LMAN (astronaut number), JOB (job type), LTYPE (sample type), NSMP and NDATA (number of samples retrieved), and LTEST (test number). These are described in more detail in section 5.

   b. Data actually retrieved include MTVOL (total volumes retrieved, --zero if no volumes retrieved), MDATE (Julian sample dates), LHIT (sample numbers), XDATE (test results in sample). Corresponding rows and columns of these four arrays correspond. That is, on this analysis, no total volume was retrieved for Julian Date 188 which had urine sample number 1443 and calcium reading of 8.2.

2. Listing of the data retrieved. The data is listed with Julian Date versus sample value separated into preflight, inflight

and post flight, respectively. Means, standard deviations, and sample sizes for each time period are listed. Data points exceeding the 95% tolerance region (above +2 standard deviations from the overall mean) are denoted with a single "+" sign, above 99% (+2.5 std. dev.) with "++", below 95% (-2 std. dev.) with "-" and below 99% (-2.5 std. dev.) with "--".

3. The means, standard deviations, and standard errors are summarized by time with the overall values also given.

4. A one-way analysis of variance is performed using preflight, inflight, and postflight as the three groups in the analysis. When only two of the groups are available, the analysis of variance is performed on the two groups. An analysis of variance table is given as well as the statement of statistical significance.

   19.070 is significant at the .1 percent level for the CDR in the example of Appendix C. This means that the F-ratio of the test is significant at .1% or that the pre-, in-, and post flight readings of calcium are different at the 99.9% significance level. Normally, a 1% or 5% is considered statistically significant.

   If the F-ratio is significant at the 5% or smaller level, then

standard contrasts of pre- to in-, in- to post-, and pre- to post flight are performed with an F-ratio and level given for each contrast. For the CDR in Appendix C, inflight is statistically significantly different from both preflight and post flight but pre- and post flight are not significantly different from each other. Examination of the values of the means would then interpret the differences as a statistically significant increase in calcium in the urine inflight with a statistically significant return to normal preflight levels post flight.

The analysis of variance was performed using the IMSL [3] routine ACRDAN. Mean squares and F-ratios are calculated by the program. Significance levels are calculated by the STATCAT [4] routines FISH and PHI. The contrasts are calculated using the IMSL [3] routine ACTRST.

5. Non-parametric tests of the Wilcoxon/Kruskal-Wallis type using preflight, inflight, and post flight as the three groups in the analysis. When only two of the groups are available, the tests are performed on the two groups. The groups are pooled, ordered in increasing order, and each observation is scored. The scores are summed by group and then combined giving the weighted score WSCORE. A level is given in the manner of the one way analysis of variance. For the CDR in Appendix C, we have a WSCORE of 38.985 which is significant at the .1% level (99.9% significance) and consequently statistically significant.

Contrasts are also performed when the test is significant at the 5% level or less. The contrasts are Wilcoxon two sample statistical tests. The ordering, Wilcoxon-type signed ranks, the ranked score W, the standard deviation and the asymptotic normal score are all given. Since the hypothesis could be either one-sided (calcium inflight is higher than calcium preflight) or two-sided (calcium inflight is different than calcium preflight), both one- and two-sided levels are given.

The routine which yields the ordered arrays and scores for both the combined analyses is the Kruskal-Wallis routine of Lee and Desu [5]. The selection of which data to use is performed by the control program.

6. Plots of the data are also produced on request (see Appendix E for request formats). The plots produced for urinary calcium on Skylab III are shown in Appendix C.

7. MODIFYING THE PROGRAM FOR SPECIAL ANALYSIS

For performing various special analyses, the program must be modified to accomodate the analysis program. Since all of the data for each astronaut is available in the subroutine STAT, usually all modifications are made only in STAT. Restructuring the data and calling of necessary routines is done from STAT.

Generally, the specialized nature of various analyses requires a programmer with some familiarity with both this retrieval and analysis program and the program to do the special analysis. As an aid to this method of program modification, all program modifications are made using a CDC system called MODIFY. By using this system it is possible to keep one copy of the standard program with all updates included and use MODIFY to create each special program as it is needed without disturbing the base program.

The subroutine STAT receives all of the data for one man at a time. The following variables and their meaning are those normally used by a special program:

LMAN(1)	Current man number
NOBS	Total data points
NNN(1)	Number data points in preflight
NNN(2)	Number data points inflight
NNN(3)	Number data points in post flight
X	Data array in date order
MDATE	Array of corresponding dates
ICASE	0 No tests or no preflight
	1 Preflight only
	2 Pre- and inflight only
	3 All
	4 Pre and post flight only

These values and arrays are good only immediately after statement 330 in STAT. Note that bad values have been edited out and certain following subroutines rearrange the data.

Two examples of program modifications are included in the program listings. The card decks used to create these modifications are listed with the resulting programs in Appendix D. The statistical details of the examples are given in [6].

8. REFERENCES

1. Leach, Carolyn S., "Skylab Endocrine-Metabolic Studies", to appear in Aerospace Medicine.

2. Giese, R. P., A Data Storage and Retrieval System for Endocrine Research, Contract NAS 9-13042, Department of Biomathematics, M. D. Anderson Hospital and Tumor Institute, 6723 Bertner, Houston, Texas, December, 1973.

3. IMSL Library 3 Reference Manual, Third Edition, International Mathematical and Statistical Library, Inc., 6200 Hillcroft, Houston, Texas, December, 1973.

4. Johnston, D. A. and Smith, E. O., STAT-CAT -- A Statistical Catalog of Subroutines and Function Subroutines, Revision 1, Computation and Analysis Division, Johnson Space Center, Program Sharing nos. C-MSC-C057, M71-50067, Febrauary, 1971.

5. Lee, E. T., and Desu, M. M., "A Computer Program for Comparing K Samples With Right-Censored Data", Computer Programs in Biomedicine, 2 (1972).

6. Johnston, Dennis A., Statistical Analysis of Skylab III, Contract NAS 9-13042, Department of Biomathematics, M.D. Anderson Hospital and Tumor Instutute, 6723 Bertner, Houston, Texas, November, 1974.

0. SAMPLE DIRECTORY

APPROVED FOR KEYPUNCHING _____ Date _____
KEYPUNCHED BY _____ Date _____

FOR	SAMPLE NUMBER	STUDY	Julian Date	MAN	TYPE	GMT	Date Collected MO	DA	YR	TOTAL Volume ml	Bag Vol ml	Bag Number	VDR	MASTER SAMPLE No. (POOL)	REMARKS
	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 27 28	29 30 31	32 33 34 35	36 37 38 39	40 41 42 43	44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	0 0														
2	0 0														
3	0 0														
4	0 0														
5	0 0														
6	0 0														
7	0 0														
8	0 0														
9	0 0														
10	0 0														
11	0 0														
12	0 0														
13	0 0														
14	0 0														
15	0 0														
16	0 0														
17	0 0														
18	0 0														
19	0 0														
20	0 0														
21	0 0														
22	0 0														
23	0 0														
24	0 0														
25	0 0														
26	0 0														
27	0 0														
28	0 0														
29	0 0														
30	0 0														
31	0 0														
32	0 0														

BIOMATH. FORM 0219-02

PSF(00-74

APPENDIX A - Data Coding Forms

1. URINE HORMONES

APPROVED FOR KEYPUNCHING_____ Date_____
Keypunched by_____ Date_____

FORM	SAMPLE NUMBER	EPI UG/TV	NOREPI UG/TV	ADH MU/TV	HYDRO UG/TV	ALDO UG/TV	17OH MG/TV	SHIAA
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	27 28 29	30 31 32

	FORM
1	0 1
2	0 1
3	0 1
4	0 1
5	0 1
6	0 1
7	0 1
8	0 1
9	0 1
10	0 1
11	0 1
12	0 1
13	0 1
14	0 1
15	0 1
16	0 1
17	0 1
18	0 1
19	0 1
20	0 1
21	0 1
22	0 1
23	0 1
24	0 1
25	0 1
26	0 1
27	0 1
28	0 1
29	0 1
30	0 1
31	0 1
32	0 1
33	0 1
34	0 1
35	0 1

BIOMATH. FORM 0219-02

PSF(00-74)

2. PLASMA HORMONES SHORT TERM

APPROVED FOR KEYPUNCHING _____ DATE _____

KEYPUNCHED BY _____ DATE _____

FORM	SAMPLE NUMBER	HYDRO ug/100 ml	ALDO pg/ml	HGH ng/ml	ANGIO mug/ml/hr	INSULIN uU/ml	T_4 ug/100 ml	ACTH	ADH	TESTOS
02	3 4 5 6	7 8 9 10	11 12 13 14	15 16 17	18 19 20 21	22 23 24 25	26 27 28	29 30 31 32	33 34 35 36 37	38 39 40 41 42

PSF(00-74)

BIOMATH. FORM 0219-02

3. PLASMA HORMONES LONG TERM

APPROVED FOR KEYPUNCHING _____ Date _____
KEYPUNCHED BY _____ Date _____

	FORM	SAMPLE NUMBER	PTH pg/ml	CAL pg/ml	VIT.D ng/ml	ACTH pg/ml	TSH
	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 27
1	0 3						
2	0 3						
3	0 3						
4	0 3						
5	0 3						
6	0 3						
7	0 3						
8	0 3						
9	0 3						
10	0 3						
11	0 3						
12	0 3						
13	0 3						
14	0 3						
15	0 3						
16	0 3						
17	0 3						
18	0 3						
19	0 3						
20	0 3						
21	0 3						
22	0 3						
	3						
24	0 3						
25	0 3						
26	0 3						
27	0 3						
28	0 3						
29	0 3						
31							
32	0 3						
33	0 3						
34	0 3						
35	0 3						

PSF(00-74)

BIOMATH. FORM 0219-02

4. URINE CHEMISTRIES

APPROVED FOR KEYPUNCHING _____ Date _____
KEYPUNCHED BY _____ Date _____

FORM	SAMPLE NUMBER	OSMO mOsm/TV	Na mEq/TV	K mEq/TV	Mg mEq/TV	PO4 mg/TV	Ca mEq/TV	Cl mEq/TV	H ml 0.1N NaOH/vol	SP GR	Creat mg/TV	Uric Acid mg/TV	
	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	27 28 29	30 31 32 33	34 35 36 37 38	39 40 41 42	43 44 45 46	47 48 49 50	51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	0 4												
2	0 4												
3	0 4												
4	0 4												
5	0 4												
6	0 4												
7	0 4												
8	0 4												
9	0 4												
10	0 4												
11	0 4												
12	0 4												
13	0 4												
14	0 4												
15	0 4												
16	0 4												
17	0 4												
18	0 4												
19	0 4												
20	0 4												
21	0 4												
22	0 4												
23	0 4												
24	0 4												
25	0 4												
26	0 4												
27	0 4												
28	0 4												
29	0 4												
30	0 4												
31	0 4												
32	0 4												
33	0 4												
34	0 4												
35	0 4												

5. BLOOD CHEMISTRIES

APPROVED FOR KEYPUNCHING _____ Date _____
KEYPUNCHED BY _____ Date _____

FORM	SAMPLE NUMBER	OSMO mOsm/l	Na mEq/l	K mEq/l	Mg mg %	PO_4 mg %	Ca mg %	Cl mEq/l	Glu mg %	TP Gm %
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	27 28 29	30 31 32 33	34 35 36
1	0.5									
2	0.5									
3	0.5									
4	0.5									
5	0.5									
6	0.5									
7	0.5									
8	0.5									
9	0.5									
10	0.5									
11	0.5									
12	0.5									
13	0.5									
14	0.5									
15	0.5									
16	0.5									
17	0.5									
18	0.5									
19	0.5									
20	0.5									
21	0.5									
	0.5									
	0.5									
24	0.5									
25	0.5									
26	0.5									
27	0.5									
28	0.5									
29	0.5									
30	0.5									
31	0.5									
32	0.5									
33	0.5									
34	0.5									
35	0.5									

BIOMATH. FORM 0218-02

PSF(00-74)

6. TRACE METALS (URINE)

APPROVED FOR KEYPUNCHING _____ Date _____
KEYPUNCHED BY _____ Date _____

	FORM	SAMPLE NUMBER	B ug/ml	Si ug/ml	Fe ug/ml	Al ug/ml	Mo ug/ml	Cu ug/ml	Zn ug/ml	Ti ug/ml	Ni ug/ml	Sr ug/ml	Cr ug/ml	Bi ug/ml	Mn ug/ml	Li ug/ml	Rb ug/ml	
	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	27 28 29 30	31 32 33 34	35 36 37 38	39 40 41 42	43 44 45 46	47 48 49 50	51 52 53 54	55 56 57 58	59 60 61 62 63 64	65 66 67 68 69	70 71 72 73 74 75 76 77 78 79 80
1	0 6																	
2	0 6																	
3	0 6																	
4	0 6																	
5	0 6																	
6	0 6																	
7	0 6																	
8	0 6																	
9	0 6																	
10	0 6																	
11	0 6																	
12	0 6																	
13	0 6																	
14	0 6																	
15	0 6																	
16	0 6																	
17	0 6																	
18	0 6																	
19	0 6																	
20	0 6																	
21	0 6																	
	6																	
	6																	
24	0 6																	
25	0 6																	
26	0 6																	
27	0 6																	
28	0 6																	
29	0 6																	
30	0 6																	
31	0 6																	
32	0 6																	
33	0 6																	
34	0 6																	
35	0 6																	

PSF(00-74)

BIOMATH. FORM 0219-02

APPROVED FOR KEYPUNCHING _____ Date _____
KEYPUNCHED BY _____ Date _____

7. 17 KETO STEROIDS (URINE)

	FORM	SAMPLE NUMBER	PD mg/TV	AND mg/TV	ETIO mg/TV	DHEA mg/TV	11=O AND mg/TV	11=O ETIO mg/TV	11OH AND mg/TV	11OH ETIO mg/TV	TOTAL Tmg/TV
	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	27 28 29 30 31	32 33 34 35 36	37 38 39 40 41	42 43 44 45 46	47 48 49 50
1	0 7										
2	0 7										
3	0 7										
4	0 7										
5	0 7										
6	0 7										
7	0 7										
8	0 7										
9	0 7										
10	0 7										
11	0 7										
12	0 7										
13	0 7										
14	0 7										
15	0 7										
16	0 7										
17	0 7										
18	0 7										
19	0 7										
20	0 7										
21	0 7										
	0 7										
	0 7										
24	0 7										
25	0 7										
26	0 7										
27	0 7										
28	0 7										
29	0 7										
30	0 7										
31	0 7										
32	0 7										
33	0 7										
34	0 7										
35	0 7										

BIOMATH. FORM 0219-02

PSF(00-74)

APPROVED FOR KEYPUNCHING __________ Date ______
KEYPUNCHED BY __________ Date ______

8. TOTAL AMINO ACIDS BASIC (URINE)

	FORM (1–2)	SAMPLE NUMBER (3–6)	Lys mg/TV (7–11)	His mg/TV (12–16)	$NH_3 \times 10^3$ mg/TV (17–21)	Arg mg/TV (22–26)	(27–80)
1	0 8						
2	0 8						
3	0 8						
4	0 8						
5	0 8						
6	0 8						
7	0 8						
8	0 8						
9	0 8						
10	0 8						
11	0 8						
12	0 8						
13	0 8						
14	0 8						
15	0 8						
16	0 8						
17	0 8						
18	0 8						
19	0 8						
20	0 8						
21	0 8						
	8						
	8						
24	0 8						
25	0 8						
26	0 8						
27	0 8						
28	0 8						
29	0 8						
30	0 8						
31	0 8						
32	0 8						
33	0 8						
34	0 8						
35	0 8						

PSF(00-74)

BIOMATH. FORM 0219-02

9. TOTAL AMINO ACIDS (URINE)
ACID AND NEUTRAL

APPROVED FOR KEYPUNCHING _____ Date _____
KEYPUNCHED BY _____ Date _____

FORM	SAMPLE NUMBER	HyP mg/TV	Asp mg/TV	Thr mg/TV	Ser mg/TV	Glu mg/TV	Pro mg/TV	Gly mg/TV	Ala mg/TV	Cys/2 mg/TV	Val mg/TV	Met mg/TV	Ile mg/TV	Leu mg/TV	Tyr mg/TV	Phe mg/TV	
. 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 27 28 29 30	31 32 33 34	35 36 37 38 39 40	41 42 43 44 45	46 47 48 49 50	51 52 53 54 55	56 57 58 59	60 61 62 63 64	65 66 67 68 69	70 71 72 73	74 75 76 77	78 79 80
1	0 9																
2	0 9																
3	0 9																
4	0 9																
5	0 9																
6	0 9																
7	0 9																
8	0 9																
9	0 9																
10	0 9																
11	0 9																
12	0 9																
13	0 9																
14	0 9																
15	0 9																
16	0 9																
17	0 9																
18	0 9																
19	0 9																
20	0 9																
21	0 9																
	9																
24	0 9																
25	0 9																
26	0 9																
27	0 9																
28	0 9																
29	0 9																
30	0 9																
31	0 9																
32	0 9																
33	0 9																
34	0 9																
35	0 9																

PSF(00-74)

BIOMATH. FORM 0219-02

10. FREE AMINO ACIDS BASIC (URINE)

APPROVED FOR KEYPUNCHING _____ Date _____
KEYPUNCHED BY _____ Date _____

FORM	SAMPLE NUMBER	Hlys mg/TV	γ-AB mg/TV	Orn mg/TV	Eth mg/TV	$NH_3 \times 10^3$ mg/TV	Lys mg/TV	1-CH_3-His mg/TV	His mg/TV	3-CH_3-His mg/TV	Ans mg/TV	Try mg/TV	Cre mg/TV	Car mg/TV	Arg mg/TV	
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 27 28 29 30	31 32 33 34 35	36 37 38 39 40 41	42 43 44 45 46 47	48 49 50 51 52 53	54 55 56 57	58 59 60 61 62	63 64 65 66 67 68 69	70 71 72 73	74 75 76 77	78 79 80
1 0																
2 1 0																
3 1 0																
4 1 0																
5 1 0																
6 1 0																
7 1 0																
8 1 0																
9 1 0																
10 1 0																
11 1 0																
12 1 0																
13 1 0																
14 1 0																
15 1 0																
16 1 0																
17 1 0																
18 1 0																
19 1 0																
20 1 0																
21 1 0																
0																
24 1 0																
25 1 0																
26 1 0																
27 1 0																
28 1 0																
29 1 0																
30 1 0																
31 1 0																
32 1 0																
33 1 0																
34 1 0																
35 1 0																

PSF(00-74)

BIOMATH. FORM 0219-02

11. FREE AMINO ACIDS (URINE)
ACID AND NEUTRAL

APPROVED FOR KEYPUNCHING _____ Date _____
KEYPUNCHED BY _____ Date _____

FO M	SAMPLE NUMBER	Pser mg/TV	Petn mg/TV	Tar mg/TV	Urea x 10^{-3} mg/TV	Hyp mg/TV	Asp mg/TV	Thr mg/TV	Ser mg/TV	Asp NH_2 mg/TV	Glu NH_2 mg/TV	Sar mg/TV	Pro mg/TV	Glu mg/TV	Cit mg/TV	Glc NH_2 mg/TV
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 27	28 29 30 31	32 33 34 35 36	37 38 39 40 41	42 43 44 45 46	47 48 49 50 51	52 53 54 55 56 57	58 59 60 61 62	63 64 65 66 67	68 69 70 71 72	73 74 75 76	77 78 79 80
1	1 1															
2	1 1															
3	1 1															
4	1 1															
5	1 1															
6	1 1															
7	1 1															
8	1 1															
9	1 1															
10	1 1															
11	1 1															
12	1 1															
13	1 1															
14	1 1															
15	1 1															
16	1 1															
17	1 1															
18	1 1															
19	1 1															
20	1 1															
	1															
	1															
23	1 1															
24	1 1															
25	1 1															
26	1 1															
27	1 1															
28	1 1															
29	1 1															
30	1 1															
31	1 1															
32	1 1															
33	1 1															
34	1 1															
35	1 1															

APPROVED FOR KEYPUNCHING _____ Date _____

KEYPUNCHED BY _____ Date _____

12. FREE AMINO ACIDS (URINE)
ACID AND NEUTRAL

FORM	SAMPLE NUMBER	Gly MG/TV	Ala mg/TV	α AA mg/TV	α AB mg/TV	Val mg/TV	Cys/a mg/TV	Cyt mg/TV	Met mg/TV	Ile mg/TV	Leu mg/TV	Tyr mg/TV	Phe mg/TV	β-Ala mg/TV	β-Aib mg/TV	
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 27	28 29 30 31 32	33 34 35 36 37	38 39 40 41 42	43 44 45 46 47	48 49 50 51 52	53 54 55 56 57	58 59 60 61 62	63 64 65 66 67	68 69 70 71 72	73 74 75 76 77	78 79 80
1	1,2															
2	1,2															
3	1,2															
4	1,2															
5	1,2															
6	1,2															
7	1,2															
8	1,2															
9	1,2															
10	1,2															
11	1,2															
12	1,2															
13	1,2															
14	1,2															
15	1,2															
16	1,2															
17	1,2															
18	1,2															
19	1,2															
20	1,2															
21	1,2															
	1,2															
24	1,2															
24	1,2															
25	1,2															
26	1,2															
27	1,2															
28	1,2															
29	1,2															
30	1,2															
31	1,2															
32	1,2															
33	1,2															
34	1,2															
35	1,2															

PSF(00-74)

BIOMATH. FORM 0219-02

13. DAILY PARAMETERS

APPROVED FOR KEYPUNCH BY _____ DATE _____
KEYPUNCHED BY _____ DATE _____

	FORM	STUDY	JULIAN DATE	BODY WEIGHT (LBS)			H_2O CONSUMPTION (ML)			PHYSICAL ACTIVITIES AND/OR ENVIRONMENTAL CONDITIONS
				CDR	PLT	SPT	CDR	PLT	SPT	
	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 27 28 29	30 31 32 33	34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	1 3									
2	1 3									
3	1 3									
4	1 3									
5	1 3									
6	1 3									
7	1 3									
8	1 3									
9	1 3									
10	1 3									
11	1 3									
12	1 3									
13	1 3									
14	1 3									
15	1 3									
16	1 3									
17	1 3									
18	1 3									
19	1 3									
20	1 3									
21	1 3									
	3									
4	1 3									
5	1 3									
6	1 3									
7	1 3									
8	1 3									
9	1 3									
30	1 3									
31	1 3									
32	1 3									
33	1 3									
34	1 3									
35	1 3									

PSF(00-74)

BIOMATH. FORM 0219-02

14. UPDATE FORM

FORM	SAMPLE NUMBER	FORM NO.	ITEM	DATA
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
	4			
2 1	4			
3 1	4			
4 1	4			
5 1	4			
6 1	4			
7 1	4			
8 1	4			
9 1	4			
10 1	4			
11 1	4			
12 1	4			
13 1	4			
14 1	4			
15 1	4			
16 1	4			
17 1	4			
18 1	4			
19 1	4			
20 1	4			
21 1	4			
22 1	4			
23 1	4			
	4			
25 1	4			
26 1	4			
27 1	4			
28 1	4			
29 1	4			
30 1	4			
31 1	4			
32 1	4			
33 1	4			
34 1	4			
35 1	4			

BIOMATH. FORM 0219-02

PSF(00-76)

15. DIET PARAMETERS

APPROVED FOR KEYPUNCHING _____ DATE _____
KEYPUNCHED BY _____ DATE _____

	FCRM	SAMPLE NUMBER	CALORIES Kcal	PROTEIN g	Ca mg	P mg	Na mg	Mg mg	K mg	H_2O ml	WEIGHT Lbs	VOL Vdr	
	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	27 28 29 30	31 32 33 34	35 36 37 38	39 40 41 42	43 44 45 46	47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	15												
2	15												
3	15												
4	15												
5	15												
6	15												
7	15												
8	15												
9	15												
10	15												
11	15												
12	15												
13	15												
14	15												
15	15												
16	15												
17	15												
18	15												
19	15												
	15												
	15												
	15												
	15												
	15												
	15												
	15												
28	15												
29	15												
30	15												
31	15												
32	15												
33	15												
34	15												
35	15												

STORATH. FORM 0219-02-15

14. A UPDATE FORM FOR FORM 13

	FORM (1–2)	JULIAN DATE (3–6)	FORM NO (7–8)	ITEM (9–10)	DATA (11–25)	STUDY (26)	(27–80)
	1 4		1 3				
2	1 4		1 3				
3	1 4		1 3				
4	1 4		1 3				
5	1 4		1 3				
6	1 4		1 3				
7	1 4		1 3				
8	1 4		1 3				
9	1 4		1 3				
10	1 4		1 3				
11	1 4		1 3				
12	1 4		1 3				
13	1 4		1 3				
14	1 4		1 3				
15	1 4		1 3				
16	1 4		1 3				
17	1 4		1 3				
18	1 4		1 3				
19	1 4		1 3				
20	1 4		1 3				
21	1 4		1 3				
	1 4		1 3				
	1 4		1 3				
	1 4		1 3				
	1 4		1 3				
	1 4		1 3				
27	1 4		1 3				
28	1 4		1 3				
29	1 4		1 3				
30	1 4		1 3				
31	1 4		1 3				
32	1 4		1 3				
33	1 4		1 3				
34	1 4		1 3				
35	1 4		1 3				

APPENDIX B - Retrieval Index

APPENDIX B

The following table lists:

INDX	A Test Number for Each Test
INDX1	Record Type (B=Blood, U=Urine)
INDX2	Name of Test
INDX3	Word Number of Record
INDX4	Form Number where Data was Punched (See Appendix E for use)
INDX5	Field Number of Input Form (See Appendix E for use)

TABLE B.1 RETRIEVAL INDEX

RETRIEVAL INDEX

INDX	INDX1	INDX2	INDX3	INDX4	INDX5	
TEST NUMBER	RECORD TYPE	NAME	WORD NUMBER OF RECORD	FORM NO.	FIELD NO.	U/L BOUND
300	B	HYDRO	2	2	3	
301	B	ALDO	3	2	4	
302	B	HGH	4	2	5	
303	B	ANGIO	5	2	6	
304	B	INSULIN	6	2	7	
305	B	T4	7	2	8	
306	B	ACTH	8	2	9	
307	B	ADH	9	2	10	
321	B	TESTOS	10	2	11	
308	B	PTH	11	3	3	
309	B	CAL	12	3	4	
310	B	VIT.C.	13	3	5	
311	B	TSH	14	3	6	
312	B	OSMO	15	5	3	
313	B	Na	16	5	4	
314	B	K	17	5	5	
315	B	Mg	18	5	6	
316	B	PO_4	19	5	7	
317	B	Ca	20	5	8	
318	B	Cl	21	5	9	
319	B	Glu	22	5	10	
320	B	TP	23	5	11	
322	(not used)					
323	(not used)					

TABLE B.1 RETRIEVAL INDEX (Continued)

RETRIEVAL INDEX

INDX	INDX1	INDX2	INDX3	INDX4	INDX5	
TEST NUMBER	RECORD TYPE	NAME	WORD NUMBER OF RECORD	FORM NO.	FIELD NO.	U/L BOUND
400	U	EPI	2	1	3	
401	U	NOREPI	3	1	4	
402	U	ADH	4	1	5	
403	U	HYDRO	5	1	6	
404	U	ALDO	6	1	7	
405	U	17OH	7	1	8	
503	U	5HIAA	8	1	9	
406	U	OSMO	9	4	3	
407	U	Na	10	4	4	
408	U	K	11	4	5	
409	U	Mg	12	4	6	
410	U	PO_4	13	4	7	
411	U	Ca	14	4	8	
412	U	Cl	15	4	9	
413	U	H	16	4	10	
414	U	Sp.Gr.	17	4	11	
415	U	Creat	18	4	12	
416	U	Uric Acid	19	4	13	
417	U	B	20	6	3	
418	U	Si	21	6	4	
419	U	Fe	22	6	5	
420	U	Al	23	6	6	
421	U	Mo	24	6	7	
422	U	Cu	25	6	8	
423	U	Zn	26	6	9	
424	U	Ti	27	6	10	
425	U	Ni	28	6	11	
426	U	Sr	29	6	12	
427	U	Cr	30	6	13	
428	U	Bl	31	6	14	
429	U	Mn	32	6	15	
430	U	Li	33	6	16	
431	U	Rb	34	6	17	
432	U	PO	35	7	3	
433	U	AND	36	7	4	
434	U	ETIO	37	7	5	
435	U	DHEA	38	7	6	
436	U	11=O AND	39	7	7	
437	U	11=O ETIO	40	7	8	
438	U	11OH AND	41	7	9	
439	U	11OHETIO	42	7	10	
440	U	TOTAL	43	7	11	

TABLE B.1 RETRIEVAL INDEX (Continued)

RETRIEVAL INDEX

INDX	INDX1	INDX2	INDX3	INDX4	INDX5	
TEST NUMBER	RECORD TYPE	NAME	WORD NUMBER OF RECORD	FORM NO.	FIELD NO.	U/L BOUND
441	U	Lys	44	8	3	
442	U	His	45	8	4	
443	U	NH_3	46	8	5	
444	U	Arg	47	8	6	
445	U	Hyp	48	9	3	
446	U	Asp	49	9	4	
447	U	Thr	50	9	5	
448	U	Ser	51	9	6	
449	U	Glu	52	9	7	
450	U	Pro	53	9	8	
451	U	Gly	54	9	9	
452	U	Ala	55	9	10	
453	U	Cys/2	56	9	11	
454	U	Val	57	9	12	
455	U	Met	58	9	13	
456	U	Ile	59	9	14	
457	U	Leu	60	9	15	
458	U	Tyr	61	9	16	
459	U	Phe	62	9	17	
460	U	Hlys	63	10	3	
461	U	γ-AB	64	10	4	
462	U	ORN	65	10	5	
463	U	ETH	66	10	6	
464	U	NH_3	67	10	7	
465	U	Lys	68	10	8	
466	U	1-CH_3-His	69	10	9	
467	U	His	70	10	10	
468	U	3-CH_3-His	71	10	11	
469	U	ANS	72	10	12	
470	U	Try	73	10	13	
471	U	Cre	74	10	14	
472	U	Car	75	10	15	
473	U	Arg	76	10	16	
474	U	Pser	77	11	3	
475	U	Petn	78	11	4	
476	U	Tar	79	11	5	
477	U	Urea	80	11	6	
478	U	Hyp	81	11	7	
479	U	Asp	82	11	8	
480	U	Thr	83	11	9	
481	U	Ser	84	11	10	
482	U	AspNH_2	85	11	11	

TABLE B.1 RETRIEVAL INDEX (Continued)

RETRIEVAL INDEX

INDX	INDX1	INDX2	INDX3	INDX4	INDX5	
TEST NUMBER	RECORD TYPE	NAME	WORD NUMBER OF RECORD	FORM NO.	FIELD NO.	U/L BOUND
483	U	$GluNH_2$	86	11	12	
484	U	Sar	87	11	13	
485	U	Pro	88	11	14	
486	U	Glu	89	11	15	
487	U	Cit	90	11	16	
488	U	$GlcNH_2$	91	11	17	
489	U	Gly	92	12	3	
490	U	Ala	93	12	4	
491	U	αAA	94	12	5	
492	U	αAB	95	12	6	
493	U	Val	96	12	7	
494	U	Cys/2	97	12	8	
495	U	Cyt	98	12	9	
496	U	Met	99	12	10	
497	U	Ile	100	12	11	
498	U	Leu	101	12	12	
499	U	Tyr	102	12	13	
500	U	Phe	103	12	14	
501	U	β-ALa	104	12	15	
502	U	β-Aib	105	12	16	
514	U	VolVDR	116	15	12	
515	U Diet	Cal	117	15	3	
516	U Diet	Prot	118	15	4	
517	U Diet	Ca	119	15	5	
518	U Diet	P	120	15	6	
519	U Diet	Na	121	15	7	
520	U Diet	Mg	122	15	8	
521	U Diet	K	123	15	9	
522	U Diet	H_2O	124	15	10	
523	U	Weight	125		11	

APPENDIX C - Sample Basic Statistical Analysis

NASA MSC ENDOCRINE DATA RETREIVAL PROGRAM. 400

```
LSTUDY      3      0      0      0      0      0      0      0      0      0
 LMAN       1      0      0      0      0      0      0      0      0
JOB JOB    DATE    3400   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0   -0
LTYPE   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
NSMP =   101          NDATA =   101
LDATE
    0
LTEST       411
MTVOL
     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      0      0      0      0      0      0      0
     0

MDATE
     188     189     190     191     192     193     194     195     196     197
     198     199     200     201     202     203     204     205     206     207
     208     209     209     210     211     212     213     214     215     216
     217     218     219     220     221     222     267     268     268     269
     270     271     272     273     274     275     276     277     278     279
     280     251     252     253     254     255     256     257     258     259
     260     261     262     263     264     265     266     281     282     237
     238     239     241     242     243     244     245     246     247     248
     249     250     266     283     223     224     225     226     227     228
     229     230     231     232     233     234     235     236     240     284
     285

LHIT
    1443    1454    1469    1475    1481    1516    1521    1528    1537    1539
    1545    1551    1562    1584    1590    1601    1602    1612    1615    1626
    1627    1683    1684    1688    1691    1694    1697    1700    1703    1706
    1709    1712    1715    1718    1721    1724    1759    1761    1771    1774
    1775    1790    1793    1798    1799    1805    1806    1809    1812    1815
    1818    1821    1824    1827    1830    1833    1836    1839    1842    1845
    1848    1851    1854    1857    1860    1863    1866    1869    1898    1899
    1902    1905    1908    1911    1914    1917    1920    1923    1926    1929
    1932    1935    1938    1941    1944    1947    1950    1953    1956    1959
    1962    1965    1968    1971    1974    1977    1980    1983    1986    1991
    1994

XDATA
     8.200000     8.000000     6.200000     8.700000     6.800000     5.000000     5.600000     6.000000
     6.700000     7.700000     7.800000     6.800000     5.900000     7.100000     7.600000     9.500000
     6.200000     6.700000     7.400000     5.700000     8.200000     7.500000    14.100000    10.300000
     6.600000     9.800000     8.300000     7.900000     9.100000     9.100000    11.000000     9.400000


     8.600000     8.800000     9.800000    10.000000     4.500000    10.200000     3.100000     9.100000
     7.000000    10.400000     8.500000     5.500000     7.200000     8.800000    11.000000     9.800000
     7.500000     7.900000     9.400000    10.800000    11.000000    10.300000    19.800000    11.900000
     8.800000    11.600000    11.200000     9.500000    10.500000    10.400000    10.300000     8.400000
    11.700000    10.700000    12.900000     6.700000     6.900000     9.700000     7.800000    11.800000
    11.100000    10.500000     9.200000    10.000000    11.800000    10.400000    11.400000     9.200000
     9.100000    10.500000    -5.000000     9.100000    10.500000    10.700000    14.300000     9.100000
    12.700000    10.800000    10.200000     8.600000    11.800000     9.600000    12.000000    12.300000
    10.100000    10.900000    11.500000     7.800000     8.800000
```

STUDY. 3 SKYLAB 3 74/04/26.

TEST.411 CA MAN. 1 CDR

TYPE. 1

JULIAN DATE	MEAN/IV	
188	5.2000	
189	6.0000	
190	6.2000	
191	6.7000	
192	6.8000	
193	5.0000	
194	5.6000	
195	6.0000	
196	6.7000	
197	7.7000	
198	7.8000	
199	6.8000	
200	5.9000	
201	7.1000	
202	7.8000	
203	9.5000	
204	6.2000	
205	6.7000	
206	7.4000	
207	5.7000	
208	8.2000	
209	7.5000	
209	14.1000	++

MEAN = 7.374 SD = 1.832 N = 23

210	10.3000	
211	6.6000	
212	9.2000	
213	8.3000	
214	7.9000	
215	9.1000	
216	9.1000	
217	11.0000	+
218	9.4000	
219	8.6000	
220	8.8000	
221	9.8000	
222	10.0000	
223	10.5000	+
224	10.7000	+
225	14.3000	++
226	9.1000	
227	12.7000	++
228	10.4000	+
229	10.2000	
230	8.6000	
231	11.3000	++
232	9.6000	

233	12.0000		++
234	12.3000		++
235	10.1000		
236	10.9000		+
237	9.7000		
238	7.8000		
239	11.8000		++
240	11.5000		+
241	11.1000		+
242	10.5000		+
243	9.2000		
244	10.0000		
245	11.3000		++
246	10.4000		+
247	11.4000		+
248	9.2000		
249	9.1000		
250	10.9000		+
251	10.6000		+
252	11.0000		+
253	10.3000		
254	19.6000		++
255	11.9000		++
256	8.2000		
257	11.6000		+
258	11.2000		+
259	9.5000		
260	10.5000		+
261	10.4000		+
262	10.3000		
263	8.4000		
264	11.7000		++
265	10.7000		+
266	-5.0000	SKIP	
266	12.9000		++
267	4.5000		
268	10.2000		
268	3.1000		--

MEAN = 10.238 SL = 2.200 N = 60

269	9.1000		
270	7.0000		
271	10.4000		+
272	9.5000		
273	5.5000		
274	7.2000		
275	8.8000		
276	11.6000		+
277	9.3000		
278	7.5000		
279	7.9000		
280	9.4000		
281	6.7000		
282	6.9000		
283	9.1000		
284	7.8000		
285	8.5000		

MEAN = 8.318 SD = 1.442 N = 17

PARAMETRIC STATISTICS.

SUMMARY

SAMPLE	N	MEAN	SD	SE
PRE	23	7.374	1.832	.382
IN	60	10.238	2.200	.284
POST	17	8.318	1.442	.350
SUM	100	9.253	2.350	.235

ANALYSIS OF VARIANCE TABLE.

	DF	SS	MS	F
TREAT	2	154.3	77.2	19.070
ERROR	97	392.5	4.0	---
TOTAL	99	546.9	5.5	---

19.070 IS SIGNIFICANT AT THE .1 PERCENT LEVEL

CONTRAST	F		
PRE-IN	23.317	IS SIGNIFICANT AT THE	.1 PERCENT LEVEL.
IN-POST	10.484	IS SIGNIFICANT AT THE	.2 PERCENT LEVEL.
PRE-POST	2.531	IS SIGNIFICANT AT THE	11.5 PERCENT LEVEL.

NON-PARAMETRIC STATISTICS.

I	OBSERVATIONS	SAMPLE	SCORES
1	3.1	2	-99.
2	4.5	2	-97.
3	5.0	1	-95.
4	5.5	3	-93.
5	5.6	1	-91.
6	5.7	1	-89.
7	5.9	1	-87.
8	6.0	1	-85.
9	6.2	1	-82.
10	6.2	1	-82.
11	6.6	2	-79.
12	6.7	1	-75.
13	6.7	1	-75.
14	6.7	3	-75.
15	6.8	1	-70.

16	6.8	1	-70.
17	6.9	3	-67.
18	7.0	3	-65.
19	7.1	1	-63.
20	7.2	3	-61.
21	7.4	1	-59.
22	7.5	1	-56.
23	7.5	3	-56.
24	7.7	1	-53.
25	7.8	3	-48.
26	7.8	1	-48.
27	7.8	1	-48.
28	7.8	2	-48.
29	7.9	3	-42.
30	7.9	2	-42.
31	8.0	1	-39.
32	8.2	1	-36.
33	8.2	1	-36.
34	8.3	2	-33.
35	8.4	2	-31.
36	8.5	3	-29.
37	8.6	2	-26.
38	8.6	2	-26.
39	8.7	1	-23.
40	8.8	2	-18.
41	8.8	2	-18.
42	8.8	3	-18.
43	8.8	3	-18.
44	9.1	2	-8.
45	9.1	3	-8.
46	9.1	2	-8.
47	9.1	2	-8.
48	9.1	2	-8.
49	9.1	3	-8.
50	9.2	2	0.
51	9.2	2	0.
52	9.4	3	4.
53	9.4	2	4.
54	9.5	1	8.
55	9.5	2	8.
56	9.6	2	11.
57	9.7	2	13.
58	9.8	2	17.
59	9.8	3	17.
60	9.8	2	17.
61	10.0	2	22.
62	10.0	2	22.
63	10.1	2	25.
64	10.2	2	28.
65	10.2	2	28.
66	10.3	2	33.
67	10.3	2	33.
68	10.3	2	33.
69	10.4	2	39.
70	10.4	3	39.
71	10.4	2	39.
72	10.5	2	45.
73	10.5	2	45.
74	10.5	2	45.
75	10.7	2	50.

76	10.7	2	50.
77	10.8	2	54.
78	10.8	2	54.
79	10.9	2	58.
80	10.9	2	58.
81	11.0	2	63.
82	11.0	2	63.
83	11.0	3	63.
84	11.1	2	67.
85	11.2	2	69.
86	11.4	2	71.
87	11.5	2	73.
88	11.6	2	75.
89	11.7	2	77.
90	11.8	2	81.
91	11.8	2	81.
92	11.8	2	81.
93	11.9	2	85.
94	12.0	2	87.
95	12.3	2	89.
96	12.7	2	91.
97	12.9	2	93.
98	14.1	1	95.
99	14.3	2	97.
100	19.8	2	99.

SAMPLE	W(I)	N(I)
1	-1259.	23.
2	1724.	60.
3	-465.	17.

B = 131171.95 T = 333106.

WSCORE = 38.985

38.985 IS SIGNIFICANT WITH PROBABILITY LESS THAN .100 PERCENT LEVEL

CONTRAST PRE-IN

I	OBSERVATIONS	SAMPLE	SCORES
1	3.1	2	-82.
2	4.5	2	-80.
3	5.0	1	-78.
4	5.6	1	-76.
5	5.7	1	-74.
6	5.9	1	-72.
7	6.0	1	-70.
8	6.2	1	-67.
9	6.2	1	-67.
10	6.6	2	-64.
11	6.7	1	-61.
12	6.7	1	-61.
13	6.8	1	-57.
14	6.8	1	-57.
15	7.1	1	-54.
16	7.4	1	-52.

51

17	7.5	1	-50.
18	7.7	1	-48.
19	7.8	2	-44.
20	7.8	1	-44.
21	7.8	1	-44.
22	7.9	2	-40.
23	8.0	1	-38.
24	8.2	1	-35.
25	8.2	1	-35.
26	8.3	2	-32.
27	8.4	2	-30.
28	8.6	2	-27.
29	8.6	2	-27.
30	8.7	1	-24.
31	8.8	2	-21.
32	8.8	2	-21.
33	9.1	2	-15.
34	9.1	2	-15.
35	9.1	2	-15.
36	9.1	2	-15.
37	9.2	2	-9.
38	9.2	2	-9.
39	9.4	2	-6.
40	9.5	1	-3.
41	9.5	2	-3.
42	9.6	2	0.
43	9.7	2	2.
44	9.8	2	5.
45	9.8	2	5.
46	10.0	2	9.
47	10.0	2	9.
48	10.1	2	12.
49	10.2	2	15.
50	10.2	2	15.
51	10.3	2	20.
52	10.3	2	20.
53	10.3	2	20.
54	10.4	2	25.
55	10.4	2	25.
56	10.5	2	30.
57	10.5	2	30.
58	10.5	2	30.
59	10.7	2	35.
60	10.7	2	35.
61	10.8	2	39.
62	10.8	2	39.
63	10.9	2	43.
64	10.9	2	43.
65	11.0	2	47.
66	11.0	2	47.
67	11.1	2	50.
68	11.2	2	52.
69	11.4	2	54.
70	11.5	2	56.
71	11.6	2	58.
72	11.7	2	60.
73	11.8	2	64.
74	11.8	2	64.
75	11.8	2	64.
76	11.9	2	68.

77	12.0	2	70.
78	12.3	2	72.
79	12.7	2	74.
80	12.9	2	76.
81	14.1	1	78.
82	14.3	2	80.
83	19.8	2	82.

W	ST. DEV.	ASYMPTOTIC WSCORE
-1089.	196.53	-5.54

-5.54 IS SIGNIFICANT AT THE .2 PERCENT LEVEL -TWO TAILED TEST
-5.54 IS SIGNIFICANT AT THE .1 PERCENT LEVEL -ONE TAILED TEST

CONTRAST IN-POST

I	OBSERVATIONS	SAMPLE	SCORES
1	3.1	2	-76.
2	4.5	2	-74.
3	5.5	3	-72.
4	6.6	2	-70.
5	6.7	3	-68.
6	6.9	3	-66.
7	7.0	3	-64.
8	7.2	3	-62.
9	7.5	3	-60.
10	7.8	3	-57.
11	7.8	2	-57.
12	7.9	3	-53.
13	7.9	2	-53.
14	8.3	2	-50.
15	8.4	2	-48.
16	8.5	3	-46.
17	8.6	2	-43.
18	8.6	2	-43.
19	8.8	3	-37.
20	8.8	2	-37.
21	8.8	3	-37.
22	8.8	2	-37.
23	9.1	2	-27.
24	9.1	2	-27.
25	9.1	2	-27.
26	9.1	3	-27.
27	9.1	2	-27.
28	9.1	3	-27.
29	9.2	2	-19.
30	9.2	2	-19.
31	9.4	3	-15.
32	9.4	2	-15.
33	9.5	2	-12.
34	9.6	2	-10.
35	9.7	2	-8.
36	9.8	2	-4.
37	9.8	3	-4.

38	9.8	2	-4.
39	10.0	2	1.
40	10.0	2	1.
41	10.1	2	4.
42	10.2	2	7.
43	10.2	2	7.
44	10.3	2	12.
45	10.3	2	12.
46	10.3	2	12.
47	10.4	2	18.
48	10.4	2	18.
49	10.4	3	18.
50	10.5	2	24.
51	10.5	2	24.
52	10.5	2	24.
53	10.7	2	29.
54	10.7	2	29.
55	10.8	2	33.
56	10.8	2	33.
57	10.9	2	37.
58	10.9	2	37.
59	11.0	2	42.
60	11.0	3	42.
61	11.0	2	42.
62	11.1	2	46.
63	11.2	2	48.
64	11.4	2	50.
65	11.5	2	52.
66	11.6	2	54.
67	11.7	2	56.
68	11.8	2	60.
69	11.8	2	60.
70	11.8	2	60.
71	11.9	2	64.
72	12.0	2	66.
73	12.3	2	68.
74	12.7	2	70.
75	12.9	2	72.
76	14.3	2	74.
77	19.8	2	76.

W	ST. DEV.	ASYMPTOTIC WSCORE
635.	162.77	3.90

3.90 IS SIGNIFICANT AT THE .2 PERCENT LEVEL -TWO TAILED TEST
3.90 IS SIGNIFICANT AT THE .1 PERCENT LEVEL -ONE TAILED TEST

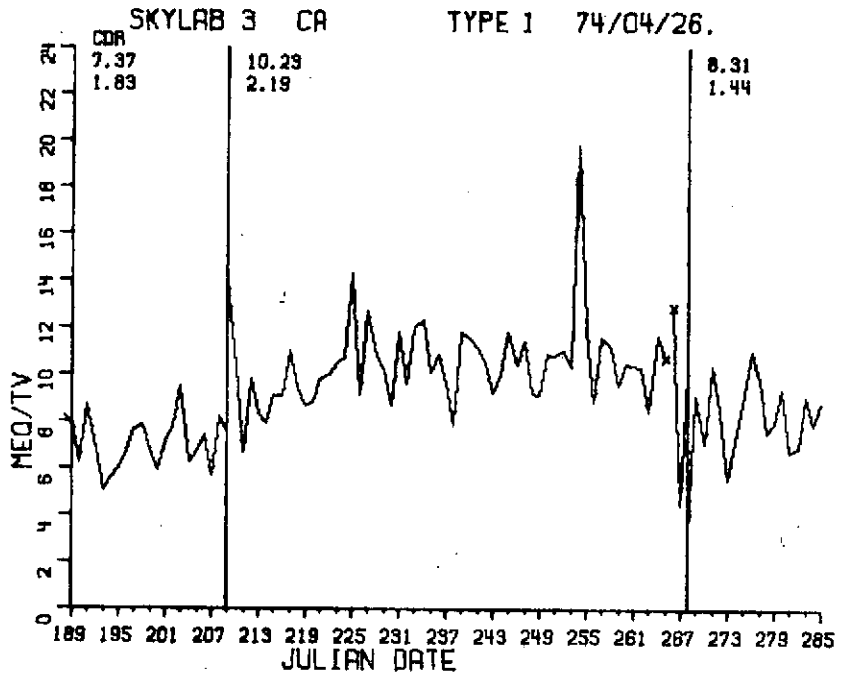
CONTRAST PRE-POST

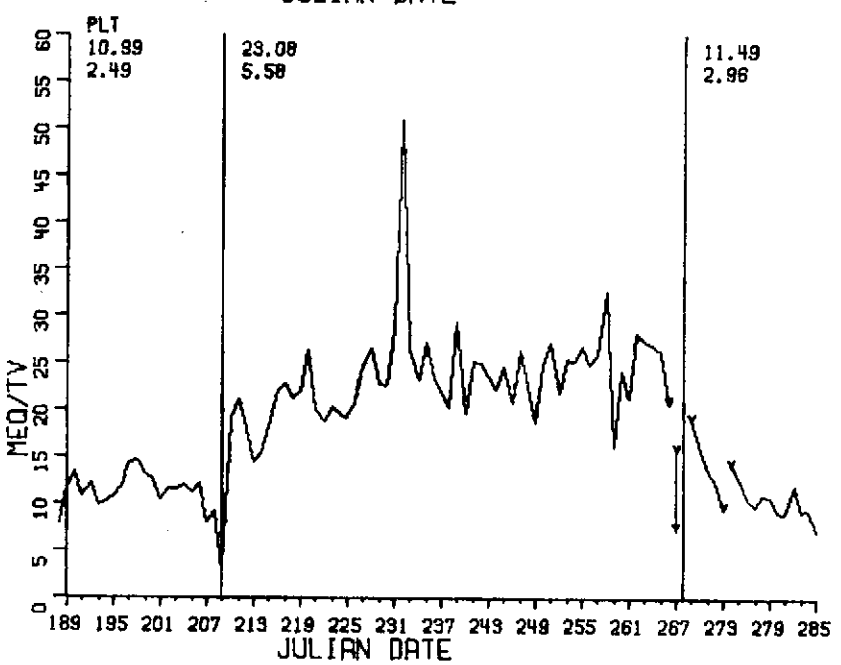
I	OBSERVATIONS	SAMPLE	SCORES
1	5.0	1	-39.
2	5.5	3	-37.
3	5.6	1	-35.
4	5.7	1	-33.

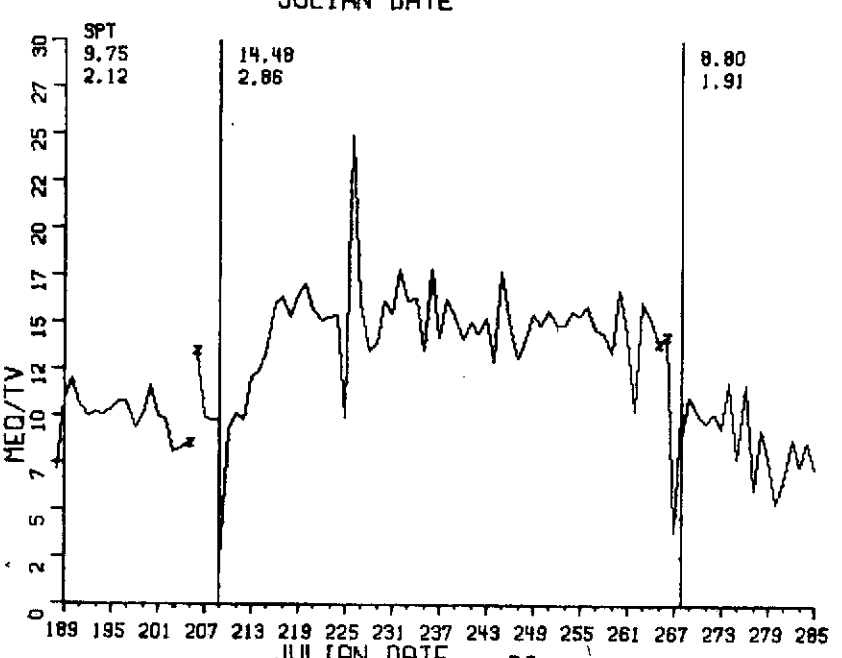
5	5.9	1	-31.
6	6.0	1	-29.
7	6.2	1	-26.
8	6.2	1	-26.
9	6.7	1	-21.
10	6.7	3	-21.
11	6.7	1	-21.
12	6.8	1	-16.
13	6.8	1	-16.
14	6.9	3	-13.
15	7.0	3	-11.
16	7.1	1	-9.
17	7.2	3	-7.
18	7.4	1	-5.
19	7.5	1	-2.
20	7.5	3	-2.
21	7.7	1	1.
22	7.8	1	5.
23	7.8	1	5.
24	7.8	3	5.
25	7.9	3	9.
26	8.0	1	11.
27	8.2	1	14.
28	8.2	1	14.
29	8.5	3	17.
30	8.7	1	19.
31	8.8	3	22.
32	8.8	3	22.
33	9.1	3	26.
34	9.1	3	26.
35	9.4	3	29.
36	9.5	1	31.
37	9.8	3	33.
38	10.4	3	35.
39	11.0	3	37.
40	14.1	1	39.

W	ST. DEV.	ASYMPTOTIC WSCORE
-170.	73.05	-2.31

-2.31 IS SIGNIFICANT AT THE 2.1 PERCENT LEVEL -TWO TAILED TEST
-2.31 IS SIGNIFICANT AT THE 1.0 PERCENT LEVEL -ONE TAILED TEST


SKYLAB 3 CA TYPE 1 74/04/26.
CDR
7.37
1.83
10.29
2.19
8.31
1.44
MEQ/TV
JULIAN DATE
189 195 201 207 213 219 225 231 237 243 249 255 261 267 273 279 285


PLT
10.99
2.49
23.08
5.58
11.49
2.96
MEQ/TV
JULIAN DATE
189 195 201 207 213 219 225 231 237 243 249 255 261 267 273 279 285


SPT
9.75
2.12
14.48
2.86
8.80
1.91
MEQ/TV
JULIAN DATE
189 195 201 207 213 219 225 231 237 243 249 255 261 267 273 279 285

APPENDIX D - Examples of Program Modifications

D.1 Program BUILD for creating and updating the data files as described in Sections 3 and 4

```
      PROGRAM BUILD(INPUT,OUTPUT,TAPE1=INPUT)                          BUILD
C RA1=SAMPLE DIRECTORY  ***  RA2=URINE FILE  ***  RA3=BLOOD FILE       BUILD
C                       ***  RA4=DAILY FILE  ***                       BUILD
      EXTERNAL ERROR1                                                  BUILD
      EXTERNAL ERROR2                                                  BUILD
      EXTERNAL ERROR3                                                  BUILD
      EXTERNAL ERROR4                                                  BUILD
      COMMON    FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)            BUILD
      COMMON /SISBUF/ IBUF(35),IUR(125),IBL(35),IDUFF(30),KEE,KEF,     BUILD
     +               INITIAL,ICODE                                     BUILD
      DIMENSION INDX(136),INDX1(136),INDX2(136),INDX3(136),            BUILD
     .  INDX4(136),INDX5(136)                                          BUILD
      DIMENSION KARD( 8), ITEMP(35), IUDATE(4), XTEMP(35)              BUILD
      INTEGER KEY(2), KYE(3)                                           BUILD
      EQUIVALENCE (ITEMP(1),XTEMP(1))                                  BUILD
      EQUIVALENCE (IUDATE(4),XUDATE)                                   BUILD
      EQUIVALENCE (KEE,KEY(2)),(KEF,KYE(3))                           BUILD
      DATA ICDT/4456/                                                  BUILD
      DATA(INDX(I),I=1,136)/400,401,402,403,404,405,503,406,407,408,409,BUILD
                                                                       BUILD
     .  410,
     1  411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,426,BUILD
     2  427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,BUILD
     3  443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,BUILD
     4  459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,BUILD
     5  475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,490,BUILD
     6  491,492,493,494,495,496,497,498,499,500,501,502,300,301,302,303,BUILD
     7  304,305,306,307,321,308,309,310,311,312,313,314,315,316,317,318,BUILD
                                                                       BUILD
     .  319,
     8  320,514,515,516,517,518,519,520,521,522,523/                  BUILD
      DATA(INDX1(I),I=1,136)/104*1RU,22*1RB,10*1RU/                    BUILD
      DATA (INDX2(I),I=1, 70)/          8HEPI      ,8HNOREPI   ,8HADH      ,BUILD
     .8HHYDRO    ,8HALDO     ,8H170H     ,8HSHIAA    ,8HOSMO     ,8HNA       ,BUILD
                                                                       BUILD
     .  8HK      ,
     .8HMG       ,8HPO4      ,8HCA       ,8HCL       ,8HH        ,8HSP.GR.   ,BUILD
     .8HCREAT    ,8HURICACID ,8HB        ,8HSI       ,8HFE       ,8HAL       ,BUILD
     .8HMO       ,8HCU       ,8HZN       ,8HTI       ,8HNI       ,8HSR       ,BUILD
     .8HCR       ,8HRL       ,8HMN       ,8HLI       ,8HRb       ,8HPD       ,BUILD
     .8HAND      ,8HETIO     ,8HDHEA     ,8H11=OANL  ,8H11=OETIO ,8H110HAND   ,BUILD
     .8H110HETIO ,8HTOTAL MG ,8HLYS      ,8HHIS      ,8HNH3      ,8HARG      ,BUILD
     .8HHYP      ,8HASP      ,8HTHR      ,8HSER      ,8HGLU      ,8HPRO      ,BUILD
     .8HGLY      ,8HALA      ,8HCYS/2    ,8HVAL      ,8HMET      ,8HILE      ,BUILD
     .8HLEU      ,8HTYR      ,8HPHE      ,8HHLYS     ,8HGAMMA-AB ,8HORN      ,BUILD
     .8HETH      ,8HNH3      ,8HLYS      ,8H1-CH-HIS ,8HHIS      ,8H3-CH-HIS/BUILD
      DATA (INDX2(I),I=71,126)/                                        BUILD
     .8HAMS      ,8HTRY      ,8HCRE      ,8HCAR      ,8HARG      ,8HPSER     ,BUILD
     .8HPETN     ,8HTAU      ,8HUREA     ,8HHYP      ,8HASP      ,8HTHR      ,BUILD
     .8HSER      ,8HASPNH2   ,8HGLUNH2   ,8HSAR      ,8HPRO      ,8HGLU      ,BUILD
     .8HCIT      ,8HGLCNH2   ,8HGLY      ,8HALA      ,8HALPHA-AA ,8HALPHA-AB ,BUILD
     .8HVAL      ,8HCYS/2    ,8HCYT      ,8HMET      ,8HILE      ,8HLEU      ,BUILD
     .8HTYR      ,8HPHE      ,8HBETA-ALA ,8HBETA-AIB ,8HHYDRO    ,8HALDO     ,BUILD
     .8HHIGH     ,8HANGIO    ,8HINSULIN  ,8HT4       ,8HACTH     ,8HADH      ,BUILD
                                                                       BUILD
     .  8HTESTOS ,
     .8HPETH     ,8HCAL      ,8HVIT.D.   ,8HTSH      ,8HOSMO     ,8HNA       ,BUILD
     .8HK        ,8HMG       ,8HPO4      ,8HCA       ,8HCL       ,8HGLU      ,BUILD
     .8HTOTAL P  /                                                     BUILD
      DATA (INDX3(I),I=1,136)/2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17, BUILD
     1  18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38, BUILD
     2  39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59, BUILD
     3  60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80, BUILD
     4  81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,   BUILD
```

```
     5101,102,103,104,105,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,BUILD
     6  20,21,22,23,116,117,118,119,120,121,122,123,124,125/              BUILD
      DATA (INDX4(I),I=1,136) /7*1,11*4,15*6,9*7,4*8,15*9,14*10,         BUILD
     1  15*11,14*12,9*2,4*3,9*5,10*15/                                    BUILD
      DATA (INDX5(I),I=1,136) /3,4,5,6,7,8,9,3,4,5,6,7,8,9,10,11,12,13,  BUILD
     1  3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,3,4,5,6,7,8,9,10,11,3,4,   BUILD
     2  5,6,3,4,                                                          BUILD
     3  5,6,7,8,9,10,11,12,13,14,15,16,17,3,4,5,6,7,8,9,10,11,12,13,     BUILD
     4  14,15,16,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,3,4,5,6,7,8,9,    BUILD
     5  10,11,12,13,14,15,16,3,4,5,6,7,8,9,10,11,3,4,5,6,  3,4,5,6,7,8,9,BUILD
     6  10,11,3,4,5,6,7,8,9,10,11,12/                                     BUILD
                                                                          BUILD
C                                                                         BUILD
C                                                                         BUILD
      IERR = 0                                                            BUILD
C     SET INITIAL 1 FOR EXISTING FILES  0 FOR NEW FILES                   BUILD
      INITIAL = 0                                                         BUILD
      CALL FILEIS(FITRA1,3LFEN,3LRA1,3LWSA, IBUF,2LRL, 350,2LKA,KEE,      BUILD
     +                    2LKP,0,2LKL,10,2LKT,1LI)                        BUILD
      CALL FILEIS(FITRA2,3LFEN,3LRA2,3LWSA,  IUR,2LRL,1250,2LKA,KEE,      BUILD
     +                    2LKP,0,2LKL,10,2LKT,1LI)                        BUILD
      CALL FILEIS(FITRA3,3LFEN,3LRA3,3LWSA,  IBL,2LRL, 350,2LKA,KEE,      BUILD
     +                    2LKP,0,2LKL,10,2LKT,1LI)                        BUILD
      CALL FILEIS(FITRA4,3LFEN,3LRA4,3LWSA,IBUFF,2LRL, 300,2LKA,KEF,      BUILD
     +                    2LKP,0,2LKL,10,2LKT,1LI)                        BUILD
      CALL STOREF(FITRA1,3LERL,0)                                         BUILD
      CALL STOREF(FITRA2,3LERL,0)                                         BUILD
      CALL STOREF(FITRA3,3LERL,0)                                         BUILD
      CALL STOREF(FITRA4,3LERL,0)                                         BUILD
      CALL STOREF(FITRA1,2LEX,ERROR1)                                     BUILD
      CALL STOREF(FITRA2,2LEX,ERROR2)                                     BUILD
      CALL STOREF(FITRA3,2LEX,ERROR3)                                     BUILD
      CALL STOREF(FITRA4,2LEX,ERROR4)                                     BUILD
      IF(INITIAL.EQ.1) GO TO 106                                          BUILD
C     CALL RANDOM(0,@RA1@,35,5,7,100S,100S,ICODE,LOC)                     BUILD
C     CALL RANDOM(0,@RA2@,125,5,7,100S,100S,ICODE,LOC)                    BUILD
C     CALL RANDOM(0,@RA3@,32,5,7,100S,100S,ICODE,LOC)                     BUILD
C     CALL RANDOM(0,@RA4@,30,5,11,100S,100S,ICODE,LOC)                    BUILD
      CALL STOREF(FITRA1,2LRB,5)                                          BUILD
      CALL STOREF(FITRA2,2LRB,5)                                          BUILD
      CALL STOREF(FITRA3,2LRB,5)                                          BUILD
      CALL STOREF(FITRA4,2LRB,5)                                          BUILD
      CALL STOREF(FITRA1,3LMNR, 350)                                      BUILD
      CALL STOREF(FITRA2,3LMNR,1250)                                      BUILD
      CALL STOREF(FITRA3,3LMNR, 350)                                      BUILD
      CALL STOREF(FITRA4,3LMNR, 300)                                      BUILD
      CALL STOREF(FITRA1,3LMRL, 350)                                      BUILD
      CALL STOREF(FITRA2,3LMRL,1250)                                      BUILD
      CALL STOREF(FITRA3,3LMRL, 350)                                      BUILD
      CALL STOREF(FITRA4,3LMRL, 300)                                      BUILD
      CALL OPENM(FITRA1,3LNEW)                                            BUILD
      CALL OPENM(FITRA2,3LNEW)                                            BUILD
      CALL OPENM(FITRA3,3LNEW)                                            BUILD
      CALL OPENM(FITRA4,3LNEW)                                            BUILD
      GO TO 40                                                            BUILD
  106 CONTINUE                                                            BUILD
C 106 CALL RANDOM(4,@RA1@,35,5,7,100S,100S,ICODE,LOC)                     BUILD
C     CALL RANDOM(4,@RA2@,125,5,7,100S,100S,ICODE,LOC)                    BUILD
C     CALL RANDOM(4,@RA3@,32,5,7,100S,100S,ICODE,LOC)                     BUILD
C     CALL RANDOM(4,@RA4@,30,5,11,100S,100S,ICODE,LOC)                    BUILD
      CALL OPENM(FITRA1,3LI-O)                                            BUILD
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      CALL OPENM(FITRA2,3LI-0)                                          BUILD
      CALL OPENM(FITRA3,3LI-0)                                          BUILD
      CALL OPENM(FITRA4,3LI-0)                                          BUILD
   40 DO 30 I=1,35                                                      BUILD
   30 ITEMP(I)=0                                                        BUILD
      ICODE = 0                                                         BUILD
      IF(INITIAL.EQ.0) ICODE = ICDT                                     BUILD
      READ(1,105) IFORM,KARD                                            BUILD
  105 FORMAT(I2,7A10,A8)                                                BUILD
      IF(EOF(1).NE.0) GO TO 99                                          BUILD
      IF(IFORM.EQ. 0) GO TO 1000                                        BUILD
      IF(IFORM.EQ.99) GO TO 99                                          BUILD
      IF(IFORM.EQ.98) GO TO 98                                          BUILD
      GO TO(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15),IFORM                  BUILD
C                                                                       BUILD
C SAMPLE DIRECTORY CARD                                                 BUILD
 1000 IEND=24                                                           BUILD
      DECODE(80,115,KARD) (ITEMP(I),I=1,12),XTEMP(13),(ITEMP(I),I=14,24)BUILD
  115 FORMAT(I4,I1,I3,I1,I2,I4,3I2,I4,I3,I5,F4.1,I4,9A4,A1)             BUILD
      GO TO 35                                                          BUILD
C     DELETE A RECORD WITH KEY AND FILE NUMBER                          BUILD
C     FFKKKKF                                                           BUILD
   98 DECODE(6,198,KARD) ITEMP(1),ITEMP(2)                              BUILD
  198 FORMAT(I4,I1)                                                     BUILD
      KEE=ITEMP(1)                                                      BUILD
      ICODE=0                                                           BUILD
      GO TO (801,802,803,804),ITEMP(2)                                  BUILD
  801 CALL DLTE(FITRA1)                                                 BUILD
      GO TO 808                                                         BUILD
  802 CALL DLTE(FITRA2)                                                 BUILD
      GO TO 808                                                         BUILD
  803 CALL DLTE(FITRA3)                                                 BUILD
      GO TO 808                                                         BUILD
  804 KEF=ITEMP(1)                                                      BUILD
      CALL DLTE(FITRA4)                                                 BUILD
  808 IF(ICODE.EQ.0) GO TO 40                                           BUILD
      PRINT 298,ITEMP(1),ITEMP(2)                                       BUILD
  298 FORMAT(* ERROR ON DELETION - KEY * I10,* FILE *,I10)              BUILD
      GO TO 310                                                         BUILD
C                                                                       BUILD
C URINE HORMONES CARD                                                   BUILD
C                                                                       BUILD
    1 IEND=8                                                            BUILD
      K=2                                                               BUILD
      DECODE(80,16,KARD) ITEMP(1),(XTEMP(I),I=2,8)                      BUILD
   16 FORMAT(I4,5F4.1,2F3.1)                                            BUILD
      GO TO 60                                                          BUILD
C                                                                       BUILD
C PLASMA HORMONES SHORT TERM CARD                                       BUILD
C                                                                       BUILD
    2 IEND = 10                                                         BUILD
      K=2                                                               BUILD
      DECODE(80,17,KARD) ITEMP(1),(XTEMP(I),I=2,10)                     BUILD
   17 FORMAT(I4,2F4.1,F3.1,F4.2,F4.1,F3.1,F4.1,F5.2,F5.1)               BUILD
      GO TO 50                                                          BUILD
C                                                                       BUILD
C PLASMA HORMONES LONG TERM CARD                                        BUILD
C                                                                       BUILD
    3 IEND=5                                                            BUILD
      K=11
```

```
      DECODE(80,18,KARD) ITEMP(1),(XTEMP(I),I=2,5)                    BUILD
   18 FORMAT(I4,F5.1,3F4.1)                                           BUILD
      GO TO 50                                                        BUILD
C                                                                     BUILD
C URINE CHEMISTRIES CARD                                              BUILD
C                                                                     BUILD
    4 IEND=12                                                         BUILD
      K=9                                                             BUILD
      DECODE(80,19,KARD) ITEMP(1),(XTEMP(I),I=2,12)                   BUILD
   19 FORMAT(I4,3F4.0,F4.1,F4.0,F3.1,F4.0,F5.1,F4.3,2F4.0)            BUILD
      GO TO 60                                                        BUILD
C                                                                     BUILD
C BLOOD CHEMISTRIES CARD                                              BUILD
C                                                                     BUILD
    5 IEND=10                                                         BUILD
      K=15                                                            BUILD
      DECODE(80,20,KARD) ITEMP(1),(XTEMP(I),I=2,10)                   BUILD
   20 FORMAT(I4,2F4.1,4F3.1,F3.0,F4.0,F3.1)                           BUILD
      GO TO 50                                                        BUILD
C                                                                     BUILD
C TRACE METALS CARD                                                   BUILD
C                                                                     BUILD
    6 IEND=16                                                         BUILD
      K=20                                                            BUILD
      DECODE(80,21,KARD) ITEMP(1),(XTEMP(I),I=2,16)                   BUILD
   21 FORMAT(I4,2F4.2,F4.3,F4.2,F4.4,F4.3,F4.2,F4.3,F4.4,F4.3,        BUILD
     .       F4.4,F4.3,F4.4,F6.4,F5.4)                                BUILD
      GO TO 60                                                        BUILD
C                                                                     BUILD
C 17KETO STEROIDS CARD                                                BUILD
C                                                                     BUILD
    7 IEND=10                                                         BUILD
      K=35                                                            BUILD
      DECODE(80,22,KARD) ITEMP(1),(XTEMP(I),I=2,10)                   BUILD
   22 FORMAT(I4,8F5.3,F4.2)                                           BUILD
      GO TO 60                                                        BUILD
C                                                                     BUILD
C TOTAL AMINO ACIDS BASIC CARD                                        BUILD
C                                                                     BUILD
    8 IEND=5                                                          BUILD
      K=44                                                            BUILD
      DECODE(80,23,KARD) ITEMP(1),(XTEMP(I),I=2,5)                    BUILD
   23 FORMAT(I4,4F5.2)                                                BUILD
      GO TO 60                                                        BUILD
C                                                                     BUILD
C TOTAL AMINO ACIDS ACID AND NEUTRAL CARD                             BUILD
C                                                                     BUILD
    9 IEND=16                                                         BUILD
      K=48                                                            BUILD
      DECODE(80,24,KARD) ITEMP(1),(XTEMP(I),I=2,16)                   BUILD
   24 FORMAT(I4,F4.2,4F5.2,F4.2,F6.2,3F5.2,F4.2,2F5.2, 2F4.2)         BUILD
      GO TO 60                                                        BUILD
C                                                                     BUILD
C FREE AMINO ACIDS BASIC CARD                                         BUILD
C                                                                     BUILD
   10 IEND=15                                                         BUILD
      K=63                                                            BUILD
      DECODE(80,25,KARD) ITEMP(1),(XTEMP(I),I=2,15)                   BUILD
   25 FORMAT(I4,F5.2,2F4.2,F6.2,  2F5.2,3F6.2,F4.2,F5.2,F7.2,2F4.2)   BUILD
      GO TO 60                                                        BUILD
```

```
C                                                                       BUILD
C FREE AMINO ACIDS ACID AND NEUTRAL CARD 1                              BUILD
C                                                                       BUILD
   11 IEND=16                                                           BUILD
      K=77                                                              BUILD
      DECODE(80,26,KARD) ITEMP(1),(XTEMP(I),I=2,16)                     BUILD
   26 FORMAT(I4,2F5.2,F6.2,F5.2,F4.2,4F5.2,F6.2,3F5.2,2F4.2)            BUILD
      GO TO 60                                                          BUILD
C                                                                       BUILD
C FREE AMINO ACIDS ACID AND NEUTRAL CARD 2                              BUILD
C                                                                       BUILD
   12 IEND=15                                                           BUILD
      K=92                                                              BUILD
      DECODE(80,27,KARD) ITEMP(1),(XTEMP(I),I=2,15)                     BUILD
   27 FORMAT(I4,F6.2,13F5.2)                                            BUILD
      GO TO 60                                                          BUILD
C                                                                       BUILD
C DAILY PARAMETERS CARD                                                 BUILD
C                                                                       BUILD
   13 IEND = 20                                                         BUILD
      DECODE(80,28,KARD) ITEMP(1),ITEMP(2),(XTEMP(I),I=3,8),            BUILD
     .                   (ITEMP(I),I=9,20)                            BUILD
   28 FORMAT(I1,I3,3F5.2,3F4.0,11A4,A3)                                 BUILD
      GO TO 70                                                          BUILD
C                                                                       BUILD
C UPDATE CARD                                                           BUILD
C                                                                       BUILD
   14 DECODE(80,29,KARD) (IUDATE(I),I=1,3),XUDATE,ISTY                  BUILD
   29 FORMAT(I4,I2,I2,F15.7,I1)                                         BUILD
      GO TO 80                                                          BUILD
C                                                                       BUILD
C DIET PARAMETER CARDS                                                  BUILD
C                                                                       BUILD
   15 IEND = 11                                                         BUILD
      K = 116                                                           BUILD
      DECODE(80,215,KARD) ITEMP(1),(XTEMP(I),I=3,11),XTEMP(2)           BUILD
  215 FORMAT(I4,F4.0,F4.1,6F4.0,F5.2,F4.0)                              BUILD
      GO TO 60                                                          BUILD
C                                                                       BUILD
C PROCESS SAMPLE DIRECTORY CARDS                                        BUILD
C                                                                       BUILD
   35 KEY(2)=ITEMP(1)                                                   BUILD
      ITEMP(25)=1RU                                                     BUILD
      IF(ITEMP(4).GT.4.AND.ITEMP(4).LT.8.OR.ITEMP(4).EQ.9) ITEMP(25)=1RBBUILD
C     CALL RANDOM(1,@RA1@,IBUF,35,KEY,37S,37S,ICODE,LOC)                BUILD
      IF(ICODE.NE.0) GO TO 37                                           BUILD
      CALL GET(FITRA1)                                                  BUILD
      IF(ICODE.NE.0) GO TO 37                                           BUILD
      PRINT 34,KEY(2)                                                   BUILD
   34 FORMAT(@ATTEMPT TO ADD EXISTING SAMPLE NO. TO SAMPLE DIR.@,I4)    BUILD
      GO TO 310                                                         BUILD
C     STOP 1                                                            BUILD
   37 IF(ICODE.EQ.ICDT) GO TO 38                                        BUILD
   36 PRINT 39,ICODE,KEY(2)                                             BUILD
   39 FORMAT(@RA1 ERROR@,O8,5X,I4)                                      BUILD
      GO TO 310                                                         BUILD
C     STOP 2                                                            BUILD
   38 CONTINUE                                                          BUILD
C  38 CALL RANDOM(2,@RA1@,ITEMP,35,KEY,36S,36S,ICODE,LOC)               BUILD
      DO 33 I=1,35                                                      BUILD
```

```
      IBUF(I) = ITEMP(I)                                                BUILD
   33 CONTINUE                                                          BUILD
      IF(ICODE.EQ.0) GO TO 1033                                         BUILD
      ICODE = 0                                                         BUILD
      CALL PUT(FITRA1)                                                  BUILD
      GO TO 1034                                                        BUILD
 1033 CALL REPLC(FITRA1)                                                BUILD
 1034 CONTINUE                                                          BUILD
      IF(ICODE.NE.0) GO TO 36                                           BUILD
      GO TO 40                                                          BUILD
C                                                                       BUILD
C PROCESS BLOOD DATA CARDS                                              BUILD
C                                                                       BUILD
   50 KEY(2)=ITEMP(1)                                                   BUILD
C     CALL RANDOM(1,@RA3@,IBL,32,KEY,53S,53S,ICODE,LOC)                 BUILD
      IF(ICODE.NE.0) GO TO 53                                           BUILD
      CALL GET(FITRA3)                                                  BUILD
      IF(ICODE.NE.0) GO TO 53                                           BUILD
   51 DO 52 J=2,IEND                                                    BUILD
      IBL(K)=ITEMP(J)                                                   BUILD
   52 K=K+1                                                             BUILD
C     CALL RANDOM(2,@RA3@,IBL,32,KEY,55S,55S,ICODE,ILOC)                BUILD
      IF(ICODE.EQ.0) GO TO 1053                                         BUILD
      ICODE = 0                                                         BUILD
      CALL PUT(FITRA3)                                                  BUILD
      GO TO 1054                                                        BUILD
 1053 CALL REPLC(FITRA3)                                                BUILD
 1054 CONTINUE                                                          BUILD
      IF(ICODE.NE.0) GO TO 55                                           BUILD
      GO TO 40                                                          BUILD
   53 IF(ICODE.EQ.ICDT) GO TO 57                                        BUILD
   55 PRINT 54,ICODE,KEY(2)                                             BUILD
   54 FORMAT(@RA3 ERROR@,O8,5X,I4)                                      BUILD
      GO TO 310                                                         BUILD
C     STOP 3                                                            BUILD
   57 IBL(1)=ITEMP(1)                                                   BUILD
      DO 58 I=2,32                                                      BUILD
   58 IBL(I)=0                                                          BUILD
      GO TO 51                                                          BUILD
C                                                                       BUILD
C PROCESS URINE DATA CARDS                                              BUILD
C                                                                       BUILD
   60 KEY(2)=ITEMP(1)                                                   BUILD
C     CALL RANDOM(1,@RA2@,IUR,125,KEY,63S,63S,ICODE,LOC)                BUILD
      IF(ICODE.NE.0) GO TO 63                                           BUILD
      CALL GET(FITRA2)                                                  BUILD
      IF(ICODE.NE.0) GO TO 63                                           BUILD
   61 DO 62 J=2,IEND                                                    BUILD
      IUR(K)=ITEMP(J)                                                   BUILD
   62 K=K+1                                                             BUILD
C     CALL RANDOM(2,@RA2@,IUR,125,KEY,65S,65S,ICODE,LOC)                BUILD
      IF(ICODE.EQ.0) GO TO 1063                                         BUILD
      ICODE = 0                                                         BUILD
      CALL PUT(FITRA2)                                                  BUILD
      GO TO 1064                                                        BUILD
 1063 CALL REPLC(FITRA2)                                                BUILD
 1064 CONTINUE                                                          BUILD
      IF(ICODE.NE.0) GO TO 65                                           BUILD
      GO TO 40                                                          BUILD
   63 IF(ICODE.EQ.ICDT) GO TO 67                                        BUILD
```

```
   65 PRINT 64,ICODE,KEY(2)                                                BUILD
   64 FORMAT(@RA2 ERROR@,O8,5X,I4)                                         BUILD
      GO TO 310                                                            BUILD
C     STOP 4                                                              BUILD
   67 IUR(1)=ITEMP(1)                                                      BUILD
      DO 68 I=2,125                                                        BUILD
   66 IUR(I)=0                                                             BUILD
      GO TO 61                                                             BUILD
C                                                                         BUILD
C PROCESS DAILY PARAMETERS                                                BUILD
C                                                                         BUILD
   70 DECODE(4,71,KARD) KYE(3)                                            BUILD
   71 FORMAT(I4)                                                          BUILD
C     CALL RANDOM(1,@RA4@,IBUF,30,KYE,77S,77S,ICODE,LOC)                  BUILD
      IF(ICODE.NE.0) GO TO 77                                             BUILD
      CALL GET(FITRA4)                                                    BUILD
      IF(ICODE.NE.0) GO TO 77                                             BUILD
      PRINT 75,KYE(2),KYE(3)                                              BUILD
   75 FORMAT(1H1,@ATTEMPT TO ADD EXISTING DAY TO DAILY FILE@,I4,5X,I4)    BUILD
      GO TO 310                                                           BUILD
C     STOP 15                                                             BUILD
   77 IF(ICODE.EQ.ICDT) GO TO 78                                          BUILD
   73 PRINT 74,ICODE,KYE(2),KYE(3)                                        BUILD
   74 FORMAT(1H1,@RA4 ERROR@,O8,5X,I4,5X,I4)                              BUILD
      GO TO 310                                                           BUILD
C     STOP 5                                                              BUILD
   78 CONTINUE                                                            BUILD
C  78 CALL RANDOM(2,@RA4@,ITEMP,30,KYE,73S,73S,ICODE,LOC)                 BUILD
      DO 79 I=1,30                                                        BUILD
      IBUFF(I) = ITEMP(1)                                                 BUILD
   79 CONTINUE                                                            BUILD
      IF(ICODE.EQ.0) GO TO 1073                                           BUILD
      ICODE = 0                                                           BUILD
      CALL PUT(FITRA4)                                                    BUILD
      GO TO 1074                                                          BUILD
 1073 CALL REPLC(FITRA4)                                                  BUILD
 1074 CONTINUE                                                            BUILD
      IF(ICODE.NE.0) GO TO 73                                             BUILD
      GO TO 40                                                            BUILD
C                                                                         BUILD
C PROCESS UPDATE CARDS                                                    BUILD
C                                                                         BUILD
   80 IF(IUDATE(2).EQ.0) GO TO 200                                        BUILD
      IF(IUDATE(2).EQ.13) GO TO 220                                       BUILD
      DO 81 I=1,136                                                       BUILD
      IF(INDX4(I).EQ.IUDATE(2).AND.INDX5(I).EQ.IUDATE(3)) GO TO 83        BUILD
   81 CONTINUE                                                            BUILD
      PRINT 82,IFORM,(IUDATE(I),I=1,3),XUDATE                             BUILD
   82 FORMAT(@NO UPDATE MATCH ON FORM AND FIELD@,5X,I2,5X,I4,2(5X,I2),    BUILD
     .  5X,F15.7)                                                         BUILD
      GO TO 310                                                           BUILD
C     STOP 6                                                              BUILD
   83 IHIT=I                                                              BUILD
      KEY(2)=IUDATE(1)                                                    BUILD
      IWRD=INDX3(IHIT)                                                    BUILD
      IF(INDX1(IHIT).EQ.1RB) GO TO 88                                     BUILD
C     CALL RANDOM(1,@RA2@,IUR,125,KEY,84S,84S,ICODE,LOC)                  BUILD
      IF(ICODE.NE.0) GO TO 84                                             BUILD
      CALL GET(FITRA2)                                                    BUILD
      IF(ICODE.NE.0) GO TO 84                                             BUILD
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      IUR(IWRD)=IUDATE(4)                                                BUILD
C     CALL RANDOM(2,@RA2@,IUR,125,KEY,85S,85S,ICODE,LOC)                 BUILD
      IF(ICODE.EQ.0) GO TO 1083                                          BUILD
      ICODE = 0                                                          BUILD
      CALL PUT(FITRA2)                                                   BUILD
      GO TO 1084                                                         BUILD
 1083 CALL REPLC(FITRA2)                                                 BUILD
 1084 CONTINUE                                                           BUILD
      IF(ICODE.NE.0) GO TO 85                                            BUILD
      GO TO 40                                                           BUILD
   84 IF(ICODE.EQ.ICDT) GO TO 87                                         BUILD
   85 PRINT 86,ICODE,KEY(2)                                              BUILD
   86 FORMAT(@RA2 ERROR ON UPDATE@,08,5X,I4)                             BUILD
      GO TO 310                                                          BUILD
C     STOP 7                                                             BUILD
   88 CONTINUE                                                           BUILD
C  88 CALL RANDOM(1,@RA3@,IBL,32,KEY,92S,92S,ICODE,LOC)                  BUILD
      IF(ICODE.NE.0) GO TO 92                                            BUILD
      CALL GET(FITRA3)                                                   BUILD
      IF(ICODE.NE.0) GO TO 92                                            BUILD
      IBL(IWRD)=IUDATE(4)                                                BUILD
C     CALL RANDOM(2,@RA3@,IBL,32,KEY,93S,93S,ICODE,LOC)                  BUILD
      IF(ICODE.EQ.0) GO TO 1093                                          BUILD
      ICODE = 0                                                          BUILD
      CALL PUT(FITRA3)                                                   BUILD
      GO TO 1094                                                         BUILD
 1093 CALL REPLC(FITRA3)                                                 BUILD
 1094 CONTINUE                                                           BUILD
      IF(ICODE.NE.0) GO TO 93                                            BUILD
      GO TO 40                                                           BUILD
   92 IF(ICODE.EQ.ICDT) GO TO 87                                         BUILD
   93 PRINT 94,ICODE,KEY(2)                                              BUILD
   94 FORMAT(@RA3 ERROR ON UPDATE@,08,5X,I4)                             BUILD
      GO TO 310                                                          BUILD
C     STOP 8                                                             BUILD
   87 PRINT 95,ICODE,KEY(2),XUDATE                                       BUILD
   95 FORMAT(@RA2 OR RA3 SAMPLE NUMBER MISSING ON UPDATE@,               BUILD
     .       08,5X,I4,3X,F15.7)                                          BUILD
      GO TO 310                                                          BUILD
C     STOP 9                                                             BUILD
  200 KEY(2) = IUDATE(1)                                                 BUILD
      IHIT = IUDATE(3) - 1                                               BUILD
C     CALL RANDOM(1,@RA1@,ITEMP,35,KEY,216S,216S,ICODE,LOC)              BUILD
      IF(ICODE.NE.0) GO TO 216                                           BUILD
      CALL GET(FITRA1)                                                   BUILD
      IF(ICODE.NE.0) GO TO 216                                           BUILD
      IBUF(IHIT) = XUDATE                                                BUILD
      IF(IHIT.EQ.13) IBUF(IHIT) = IUDATE(4)                              BUILD
      IF(IHIT .EQ. 13) ITEMP(IHIT) = IUDATE(4)                           BUILD
C     CALL RANDOM(2,@RA1@,ITEMP,35,KEY,217S,217S,ICODE,LOC)              BUILD
      IF(ICODE.EQ.0) GO TO 1213                                          BUILD
      ICODE = 0                                                          BUILD
      CALL PUT(FITRA1)                                                   BUILD
      GO TO 1214                                                         BUILD
 1213 CALL REPLC(FITRA1)                                                 BUILD
 1214 CONTINUE                                                           BUILD
      IF(ICODE.NE.0) GO TO 217                                           BUILD
      GO TO 40                                                           BUILD
  216 IF(ICODE.EQ.ICDT) GO TO 230                                        BUILD
  217 PRINT 218,ICODE,KEY(2)                                             BUILD
```

```
  218 FORMAT(@RA1 ERROR ON UPDATE@,08,5X,I4)                              BUILD
      GO TO 310                                                           BUILD
C     STOP 12                                                             BUILD
  220 KYE(3) = IUDATE(1) + 1000 * ISTY                                    BUILD
      IHIT = IUDATE(3) - 1                                                BUILD
C     CALL RANDOM(1,@RA4@,ITEMP,30,KYE,221S,221S,ICODE,LOC)               BUILD
      IF(ICODE.NE.0) GO TO 221                                            BUILD
      CALL GET(FITRA4)                                                    BUILD
      IF(ICODE.NE.0) GO TO 221                                            BUILD
      IBUFF(IHIT) = IUDATE(4)                                             BUILD
C     CALL RANDOM(2,@RA4@,ITEMP,30,KYE,222S,222S,ICODE,LOC)               BUILD
      IF(ICODE.EQ.0) GO TO 1223                                           BUILD
      ICODE = 0                                                           BUILD
      CALL PUT(FITRA4)                                                    BUILD
      GO TO 1224                                                          BUILD
 1223 CALL REPLC(FITRA4)                                                  BUILD
 1224 CONTINUE                                                            BUILD
      IF(ICODE.NE.0) GO TO 222                                            BUILD
      GO TO 40                                                            BUILD
  221 IF(ICODE.EQ.ICDT) GO TO 240                                         BUILD
  222 PRINT 223,ICODE,KYE(2),KYE(3)                                       BUILD
  223 FORMAT(1H1,@RA4 ERROR ON UPDATE@,08,5X,I4,5X,I4)                    BUILD
      GO TO 310                                                           BUILD
C     STOP 13                                                             BUILD
  240 PRINT 241,ICODE,KYE(2),KYE(3),XUDATE                                BUILD
  241 FORMAT(@RA4 DAY AND STUDY MISSING ON UPDATE@,                       BUILD
     .       08,5X,I4,5X,I4,5X,F15.7)                                     BUILD
      GO TO 310                                                           BUILD
C     STOP 14                                                             BUILD
  230 PRINT 231,ICODE,KEY(2),XUDATE                                       BUILD
  231 FORMAT(@RA1 SAMPLE NUMBER MISSING ON UPDATE@,08,5X,I4,5X,F15.7)     BUILD
      GO TO 310                                                           BUILD
C     STOP 16                                                             BUILD
C                                                                         BUILD
C CLOSE FILES AND SAVE                                                    BUILD
C                                                                         BUILD
   99 CONTINUE                                                            BUILD
C  99 CALL RANDOM(3,@RA1@)                                                BUILD
C     CALL RANDOM(3,@RA2@)                                                BUILD
C     CALL RANDOM(3,@RA3@)                                                BUILD
C     CALL RANDOM(3,@RA4@)                                                BUILD
      CALL CLOSEM(FITRA1)                                                 BUILD
      CALL CLOSEM(FITRA2)                                                 BUILD
      CALL CLOSEM(FITRA3)                                                 BUILD
      CALL CLOSEM(FITRA4)                                                 BUILD
      PRINT 300                                                           BUILD
  300 FORMAT(17H0UPDATE COMPLETE.)                                        BUILD
      STOP 10                                                             BUILD
  100 PRINT 101,ICODE, JCODE                                              BUILD
  101 FORMAT(@ OPEN OR UPDATE BEGIN ERROR@,5X,08,2X,08)                   BUILD
C     STOP 11                                                             BUILD
  310 PRINT 400,IFORM,KARD                                                BUILD
  400 FORMAT(I4,19A4,A2)                                                  BUILD
      GO TO 40                                                            BUILD
      END                                                                 BUILD
      SUBROUTINE ERROR1                                                   ERROR1
      COMMON    FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)               ERROR1
      COMMON /SISBUF/ IBUF(35),IUR(125),IBL(35),IBUFF(30),KEE,KEF,        ERROR1
     +                INITIAL,ICODE                                      ERROR1
      ICODE = IFETCH(FITRA1,3LIRS)                                        ERROR1
```

```
      CALL STOREF(FITRA1,3LIRS,0)                                       ERROR1
      IF(ICODE.EQ.0.OR.ICODE.EQ.445B) RETURN                            ERROR1
      PRINT 100,ICODE                                                   ERROR1
  100 FORMAT(*  RA1 ERROR*,O10)                                         ERROR1
      RETURN                                                            ERROR1
      END                                                               ERROR1
      SUBROUTINE ERROR2                                                 ERROR2
      COMMON     FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)            ERROR2
      COMMON /SISBUF/ IBUF(35),IUR(125),IBL(35),IBUFF(30),KEE,KEF,      ERROR2
     +               INITIAL,ICODE                                      ERROR2
      ICODE = IFETCH(FITRA2,3LIRS)                                      ERROR2
      CALL STOREF(FITRA2,3LIRS,0)                                       ERROR2
      IF(ICODE.EQ.0.OR.ICODE.EQ.445B) RETURN                            ERROR2
      PRINT 100,ICODE                                                   ERROR2
  100 FORMAT(*  RA2 ERROR*,O10)                                         ERROR2
      RETURN                                                            ERROR2
      END                                                               ERROR2
      SUBROUTINE ERROR3                                                 ERROR3
      COMMON     FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)            ERROR3
      COMMON /SISBUF/ IBUF(35),IUR(125),IBL(35),IBUFF(30),KEE,KEF,      ERROR3
     +               INITIAL,ICODE                                      ERROR3
      ICODE = IFETCH(FITRA3,3LIRS)                                      ERROR3
      CALL STOREF(FITRA3,3LIRS,0)                                       ERROR3
      IF(ICODE.EQ.0.OR.ICODE.EQ.445B) RETURN                            ERROR3
      PRINT 100,ICODE                                                   ERROR3
  100 FORMAT(*  RA3 ERROR*,O10)                                         ERROR3
      RETURN                                                            ERROR3
      END                                                               ERROR3
      SUBROUTINE ERROR4                                                 ERROR4
      COMMON     FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)            ERROR4
      COMMON /SISBUF/ IBUF(35),IUR(125),IBL(35),IBUFF(30),KEE,KEF,      ERROR4
     +               INITIAL,ICODE                                      ERROR4
      ICODE = IFETCH(FITRA4,3LIRS)                                      ERROR4
      CALL STOREF(FITRA4,3LIRS,0)                                       ERROR4
      IF(ICODE.EQ.0.OR.ICODE.EQ.445B) RETURN                            ERROR4
      PRINT 100,ICODE                                                   ERROR4
  100 FORMAT(*  RA4 ERROR*,O10)                                         ERROR4
      RETURN                                                            ERROR4
      END                                                               ERROR4
```

D.2 Program TYPLOK is used to determine the content and integrity of the data files See Sections 3 and 4

```
      PROGRAM TYPELOK(INPUT,OUTPUT)
      EXTERNAL ERROR
      COMMON FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35),IERR,IFILE
      COMMON UTYPE(125,5,5),BTYPE(25,4,5)
      COMMON MAN(3,125)
      INTEGER UTYPE,BTYPE,TYPE
      DIMENSION IBUF(35),XBUF(35),IBUF2(125),XBUF2(125)
      DIMENSION IBUF3(35),XBUF3(35)
      EQUIVALENCE (IBUF3(1),XBUF3(1))
      EQUIVALENCE(IBUF(1),XBUF(1)),(IBUF2(1),XBUF2(1))
      CALL FILEIS(FITRA1,3LIFN,3LRA1,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5
     +           ,3LMRL, 350,3LMNR, 350,3LWSA, IBUF)
      CALL STOREF(FITRA1,3LERL,0)
      CALL STOREF(FITRA1,2LEX,  ERROR)
      CALL OPENM(FITRA1,3LI-0)
      CALL REWND(FITRA1)
      CALL FILEIS(FITRA2,3LIFN,3LRA2,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5
     +           ,3LMRL,1250,3LMNR,1250,3LWSA,IBUF2)
      CALL STOREF(FITRA2,3LERL,0)
      CALL STOREF(FITRA2,2LEX,  ERROR)
      CALL OPENM(FITRA2,5LINPUT)
      CALL FILEIS(FITRA3,3LIFN,3LRA3,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5
     +           ,3LMRL, 350,3LMNR, 350,3LWSA, IBUF3)
      CALL STOREF(FITRA3,3LERL,0)
      CALL STOREF(FITRA3,2LEX,  ERROR)
      CALL OPENM(FITRA3,5LINPUT)
      DO 11 I=1,3
      DO 11 J=1,125
   11 MAN(I,J)=0
      DO 4 M=1,5
      DO 5 J=1,125
      DO 5 I=1,5
    5 UTYPE(J,I,M)=0
      DO 6 J=1,25
      DO 6 I=1,4
    6 BTYPE(J,I,M)=0
    4 CONTINUE
      KSAV=0
      ISTART=0
    9 IERR=0
      IFILE=1
      CALL GETN(FITRA1)
      IF(IFETCH(FITRA1,2LFP).EQ.100B) GO TO 40
      IF(IERR.NE.0) STOP 1
      IF(KEY.EQ.IBUF(1)) GO TO 10
      PRINT 601,KEY,IBUF(1)
  601 FORMAT(* MISMATCH KEY *,O20,1X,O20)
      GO TO 9
   10 IF(ISTART.NE.0) GO TO 20
      ISTART=1
      GO TO 21
   20 IF(KSAV +1.EQ.KEY) GO TO 22
   21 PRINT 602,KSAV,KSAV,KEY,KEY
  602 FORMAT(* KEY SEQUENCE - PREVIOUS *,O5,I5,* CURRENT *,O5,I5)
   22 KSAV=KEY
C     FIND WHICH RECORD AND FILE TO READ
      TYPE=IBUF(5)
      IF((IBUF(5).GE.1.AND.IBUF(5).LE.4).OR.IBUF(5).EQ.8) GO TO 25
      IF((IBUF(5).GE.5.AND.IBUF(5).LE.7).OR.IBUF(5).EQ.9) GO TO 35
      PRINT 603
```

```
  603 FORMAT(* INCORRECT TYPE - ASSUME URINE(RA2)*)
      PRINT 609,KEY,(IBUF(I),I=1,5)
  609 FORMAT(1H+,T40,*KEY=*,O5,* SAMPLE=*,I5,* STUDY=*,I2,* DATE=*,I4,*
     1MAN=*,I2,* TYPE=*,I2)
   25 CONTINUE
      MFLAG=0
      IERR=0
      IFILE=2
      CALL GET(FITRA2)
      IF(IERR.NE.0) GO TO 7
      IF(TYPE.LT.1.OR.TYPE.GT.9)GO TO 26
      IF(TYPE.EQ.8) TYPE=5
      M=IBUF(2)
      IF(M.GE.1.AND.M.LE.5) GO TO 12
      PRINT 703,(IBUF(I),I=1,5)
  703 FORMAT(* INCORRECT STUDY*,5I10)
      GO TO 26
   12 CONTINUE
C     PRIME CREW TEST
      IF(IBUF(4).GT.3) GO TO 26
      DO 3 I=2,125
C     USABLE DATA TEST
      IF(XBUF2(I).EQ.0) GO TO 3
      IF(M. EQ.3) MAN(IBUF(4),I)=MAN(IBUF(4),I)+1
      IF(XBUF2(I).LT.0..AND.XBUF2(I).NE.-2.) GO TO 3
      UTYPE(I,TYPE,M)=UTYPE(I,TYPE,M)+1
    3 CONTINUE
      GO TO 26
    7 CONTINUE
      IF(IERR.EQ.445B) GO TO 27
      PRINT 604,IERR,KEY
  604 FORMAT(* RA2 ERROR *,O20,* KEY *,O20)
      GO TO 26
   27 PRINT 605
      PRINT 609,KEY,(IBUF(I),I=1,5)
      MFLAG=1
  605 FORMAT(* MISSING URINE RECORD *)
   26 IERR=0
      IFILE=3
      CALL GET(FITRA3)
      IF(IERR.EQ.445B) GO TO 9
      IF(IERR.NE.0) PRINT 606,IERR
  606 FORMAT(* ERROR ON RA3 AT 26 *,O20)
      PRINT 607,KEY,IBUF3(1),(XBUF3(I),I=2, 35)
  607 FORMAT(* BLOOD RECORD IN URINE TYPE *,O20,I5/4(4X,10F12.3/))
      IF(MFLAG.EQ.0) PRINT 620,KEY,IBUF2(1),(XBUF2(I),I=2,125)
  620 FORMAT(* URINE RECORD *,O20,I5/13(4X,10F12.3/))
      GO TO 9
   35 IERR=0
      IFILE=3
      MFLAG=0
      CALL GET(FITRA3)
      IF(IERR.NE.0) GO TO 8
      IF(TYPE.LT.1.OR.TYPE.GT.9) GOT O 36
      TYPE=IBUF(5)-4
      IF(TYPE.EQ.5) TYPE=4
      M=IBUF(2)
      IF(M.GE.1.AND.M.LE.5) GO TO 13
      PRINT 703,(IBUF(I),I=1,5)
      GO TO 36
```

```
   13 CONTINUE
C     PRIME CREW TEST
      IF(IBUF(4).GT.3) GO TO 36
      DO 2 I=2,25
C     USEABLE DATA TEST
      IF(XBUF3(I).EQ.0) GO TO 2
      IF(XBUF3(I).LT.0.AND.YBUF3(I).NE.-2.) GO TO 2
      BTYPE(I,TYPE,M)=BTYPE(I,TYPE,M)+1
    2 CONTINUE
      GO TO 36
    8 CONTINUE
      IF(IERR.EQ.445B) GO TO 37
      PRINT 614,IERR,KEY
  614 FORMAT(* RA3 ERROR *,O20,* KEY *,O20)
      GO TO 36
   37 PRINT 615
      PRINT 609,KEY,(IBUF(I),I=1,5)
  615 FORMAT(* MISSING BLOOD RECORD *)
      MFLAG=1
   36 IERR=0
      IFILE=2
      CALL GET(FITRA2)
      IF(IERR.EQ.445B) GO TO 9
      IF(IERR.NE.0) PRINT 616,IERR
  616 FORMAT(* ERROR ON RA2 AT 36 *,O20)
      PRINT 617,KEY,IBUF2(1),(XBUF2(I),I=2,125)
  617 FORMAT(* URINE RECORD IN BLOOD TYPE *,O20,I5/13(4X,10F12.3/))
      IF(MFLAG.EQ.0) PRINT 621,KEY,IBUF3(1),(XBUF3(I),I=2,35)
  621 FORMAT(* BLOOD RECORD *,O20,I5/4(4X,10F12.3/))
      GO TO 9
   40 CONTINUE
      PRINT 602,KEY,KEY
      CALL REWND(FITRA2)
   41 IERR=0
      IFILE=2
      CALL GETN(FITRA2)
      IF(IFETCH(FITRA2,2LFP).EQ.100B) GO TO 50
      IERR=0
      IFILE=1
      CALL GET(FITRA1)
      IF(IERR.EQ.445B) PRINT 618,KEY,KEY
  618 FORMAT(* MISSING RA1 RECORD FROM RA2 *,O20,I10)
      GO TO 41
   50 CALL REWND(FITRA3)
   51 IERR=0
      IFILE=3
      CALL GETN(FITRA3)
      IF(IFETCH(FITRA3,2LFP).EQ.100B) GO TO 60
      IERR=0
      IFILE=1
      CALL GET(FITRA1)
      IF(IERR.EQ.445B) PRINT 619,KEY,KEY
  619 FORMAT(* MISSING RA1 RECORD FROM RA3 *,O20,I10)
      GO TO 51
   60 CONTINUE
      PRINT 801,(I,I=1,5)
  801 FORMAT(1H1,T46,*URINE DATA SAMPLE COUNT*/T55,*STUDY*/T20,92(*-*)/
     15(20X,I2)/5(18X,*TYPE*)/5X,5(2X,20(*-*))/6X,5(*    1   2   3   4
     2 5 *)/* TEST*)
      DO 62 I=2,125
```

```
      K=I+398
      IF(I.EQ.8) K=503
      IF(I.GT.8.AND.I.LT.106) K=K-1
      PRINT 701,K,((UTYPE(I,J,M),J=1,5),M=1,5)
   62 CONTINUE
  701 FORMAT((I5,5(2X,5I4)))
      PRINT 610
  610 FORMAT(*1  COUNT OF URINE SAMPLE DATA POINTS BY MAN AND TEST*)
      PRINT 608,MAN
  608 FORMAT(3I5,2X,3I5,2X,3I5,2X,3I5,2X,3I5)
      PRINT 802,(I,I=1,5)
  802 FORMAT(1H1,T44,*BLOOD DATA SAMPLE COUNT*/T53,*STUDY*/T17,76(*-*)/
     15(16X,I2)/5(14X,*TYPE*)/5X,5(2X,16(*-*))/6X,5(*    5   6   7   9 *
     2)/* TEST*)
      DO 61 I=2,25
      K=I+298
      IF(I.EQ.10) K=321
      IF(I.GT.10.AND.I.LT.24) K=K-1
      PRINT 702,K,((BTYPE(I,J,M),J=1,4),M=1,5)
   61 CONTINUE
  702 FORMAT((I5,5(2X,4I4)))
      CALL CLOSEM(FITRA1)
      CALL CLOSEM(FITRA2)
      CALL CLOSEM(FITRA3)
      END
      SUBROUTINE ERROR
      COMMON FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35),IERR,IFILE
      GO TO (100,200,300,400) IFILE
  100 IERR = IFETCH(FITRA1,3LIRS)
      CALL STOREF(FITRA1,3LIRS,0)
      GO TO 500
  200 IERR = IFETCH(FITRA2,3LIRS)
      CALL STOREF(FITRA2,3LIRS,0)
      GO TO 500
  300 IERR = IFETCH(FITRA3,3LIRS)
      CALL STOREF(FITRA3,3LIRS,0)
      GO TO 500
  400 IERR = IFETCH(FITRA4,3LIRS)
      CALL STOREF(FITRA4,3LIRS,0)
  500 IF(IERR.EQ.0) RETURN
C     PRINT 601,IFILE,IERR
  601 FORMAT(* RA*,I1,* ERROR*,O10)
      RETURN
      END
```

D.3 Program RETD for performing a basic analysis of the data

```
      PROGRAM  RETD(INPUT,OUTPUT)                                          RETD
C     MAIN PROGRAM                                                         RETD
      COMMON XDATA(100),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP           REDUCE
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)             RETD
      COMMON /HITBLK/LHIT(100),MDATE(100),ISAV(16),MTVOL(100)              REDUCE
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),      RETD
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA              RETD
      COMMON/RG/ DATESC(3,5),IVV(4), SSMEAN(3), SSIGMA(3), NNN(3),SSE(3)RETD
      COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,         RETD
     +                MFUNC,IFILE,IERR                                     RETD
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)              RETD
C     START,LAUNCH,SPLASH DOWN                                             JAN25
      DATA DATESC/180.,208.,265.,114.,144.,172.,189.,209.,268.,            DEC10
     X 283.,320.,404.,  290.,342.,354./                                    DEC10
C     SL4 SPLASH DOWN IS 39  FEB 8 1974                                    DEC10
      IOPLOT = 0                                                           RETD
      KSKIP = 0                                                            RETD
C     IFFF IS LOOP CONTROL TO ALLOW OVERLAY                                JAN25
      IFFF = -1                                                            RETD
    1 CALL RET                                                             RETD
      CALL STAT                                                            RETD
      GO TO 1                                                              RETD
      END                                                                  RETD
      SUBROUTINE LINE(X,Y,N,M,L,LL)                                        LINE
C     X IS THE DATE. (IMDEPENDENT VARIABLE)                                LINE
C     Y IS THE DATA. (DEPENDENT VARIABLE)                                  LINE
C     IFLAG = 13   LAST POINT NOT VALID                                    LINE
C     JFLAG = 13   LAST POINT VALID                                        LINE
C     ISFLAG IS THE NUMBER OF POINTS PLOTED.                               LINE
      DIMENSION X(1), Y(1)                                                 LINE
      DATA D/0.04/                                                         LINE
      IFLAG = 0                                                            LINE
      JFLAG = 0                                                            LINE
      ISFLAG = 0                                                           LINE
      IP = 3                                                               LINE
      IF(N - 1) 9,1,1                                                      LINE
    1 DO 6 I=1,N                                                           LINE
      II = I*M - M + 1                                                     LINE
      XP = (X(II) - X(N + M))/X(N + 2*M)                                   LINE
      YP = (Y(II) - Y(N + M))/Y(N + 2*M)                                   LINE
C     CHECK TO SEE IF POINT VALUE IS VALID.                                LINE
      IF(Y(II)) 3,13,23                                                    LINE
    3 IY = -Y(II) + 0.5                                                    LINE
      IF(IY - 2) 13,14,13                                                  LINE
C     THIS IS ZERO ON THE GRAPH.                                           LINE
   14 YP = -Y(N+1)/Y(N+2*M)                                                LINE
      GO TO 23                                                             LINE
C     INVALID POINT.                                                       LINE
   13 IFLAG = 13                                                           LINE
      GO TO 6                                                              LINE
C     VALID POINT.  CHECK FOR CONTINUOUS DATE.                             LINE
   23 JFLAG = 13                                                           LINE
      IF(ISFLAG) 25,25,24                                                  LINE
   24 IF(X(II) - X(II-M) - 1.5) 25,35,35                                   LINE
   25 IF(IFLAG) 35,26,35                                                   LINE
C     PLOT THIS PEN DOWN.                                                  LINE
   26 CALL PLOT(XP,YP,IP)                                                  LINE
      ISFLAG = ISFLAG + 1                                                  LINE
      IP = 2                                                               LINE
      GO TO 6                                                              LINE
```

```
C       DISCONTINUOUS GRAPH, PLOT AN X ON EACH END OF LINE.                    LINE
   35   CALL WHERE(U,V,F)                                                      LINE
        IFLAG = 0                                                              LI
        KFLAG = JFLAG*ISFLAG                                                   LI
        IF(KFLAG) 16,17,16                                                     LINE
   16    CALL SYMBOL(U,V,D,LL,0.0,-1)                                          LINE
        ISFLAG = ISFLAG + 1                                                    LINE
   17   CALL SYMBOL(XP,YP,D,LL,0.0,-1)                                         LINE
        IP = 2                                                                 LINE
        ISFLAG = ISFLAG + 1                                                    LINE
    6   CONTINUE                                                               LINE
        KFLAG = IFLAG*JFLAG                                                    LINE
        IF(KFLAG) 9,9,29                                                       LINE
   29   CALL WHERE(U,V,F)                                                      LINE
        CALL SYMBOL(U,V,D,LL,0.0,-1)                                           LINE
        ISFLAG = ISFLAG + 1                                                    LINE
    9   RETURN                                                                 LINE
        END                                                                    LINE
        SUBROUTINE AXIS (X,Y,BCD,NC,SIZE,THETA,YMIN,DY,NDEC,NLAB,NTIC)         AXIS
        DIMENSION G(2), H(11)                                                  AXIS
        DATA G/.8,.56/                                                         AXIS
        DATA H/.56,.4,.28,.2,.14,.1,.07,.05,.035,.025,.0175/                   AXIS
        AC = NC                                                                AXIS
        SIG=SIGN(1.0,AC)                                                       AXIS
    2   NAC=IABS(NC)                                                           AXIS
        TH=THETA*0.017453294                                                   AXIS
        IF(NLAB.LE.0) NLAB = 1                                                 AXIS
        IF(NTIC.LE.0) NTIC = 1                                                 AXIS
        FNLAB = NLAB                                                           AX
        N = SIZE*FNLAB + 0.1                                                   AX
C       N = SIZE + 0.50                                                        AXIS
        CTH = COS  (TH)                                                        AXIS
        STH = SIN  (TH)                                                        AXIS
        CTN = CTH/FNLAB                                                        AXIS
        STN = STH/FNLAB                                                        AXIS
        TN = N                                                                 AXIS
        N1 = N + 1                                                             AXIS
        N2 = N1/2                                                              AXIS
        ADY=ABS(DY/FNLAB)                                                      AXIS
C       ADY=ABS(DY)                                                            AXIS
        STAT=YMIN                                                              AXIS
        EXP = 0.0                                                              AXIS
        IF ( ADY ) 9,18,9                                                      AXIS
    9   IF ( ADY -100.0 ) 10,12,12                                             AXIS
   12   ADY = ADY / 10.0                                                       AXIS
        STAT=STAT/10.0                                                         AXIS
        EXP = EXP + 1.0                                                        AXIS
        GO TO 9                                                                AXIS
   14   ADY = ADY * 10.0                                                       AXIS
        STAT=STAT*10.0                                                         AXIS
        EXP = EXP - 1.0                                                        AXIS
   10   IF ( ADY - 1.00 ) 14,18,18                                             AXIS
C  10   IF ( ADY - 0.01 ) 14,18,18                                             AXIS
C  18   XA = X   - (.20 * SIG   - .05) *STH - .0857 * CTH                      AXIS
   18   XA = X   - (H(NLAB+1)  * SIG   - H(NLAB+5)) *STH - .0857 * CTH         AX
C       YA = Y   + (.20 * SIG   - .05) * CTH - .0857 * STH                     AX
        YA = Y   + (H(NLAB+1) * SIG   - H(NLAB+5)) * CTH - .0857 * STH         AXIS
        I = 0                                                                  AXIS
   25   I = I + 1                                                              AXIS
C       CALL NUMBER (XA,YA,0.1,STAT,THETA,2)                                   AXIS
```

```
      CALL NUMBER (XA,YA,H(NLAB+3),STAT,THETA,NDEC)                          AXIS
      STAT=STAT+SIGN(ADY,DY)                                                 AXIS
C     XA = XA + CTH                                                          AXIS
      XA = XA + CTN                                                          AXIS
C     YA = YA + STH                                                          AXIS
      YA = YA + STN                                                          AXIS
      IF(I - N2) 25,31,26                                                    AXIS
   26 IF(I - N1) 25,60,60                                                    AXIS
   31 TNC = NAC + 7                                                          AXIS
C     XC = X + (SIZE / 2.0 -.06 * TNC)*CTH - (-.07 + SIG *.36)* STH          AXIS
      XC = X + (SIZE / 2.0 -H(NLAB+4) * TNC)*CTH                             AXIS
     1- (-H(NLAB+4) + SIG *(H(NLAB) + H(NLAB+3)))* STH                       AXIS
C     YC = Y + (SIZE / 2.0 -.06 * TNC)*STH + (-.07 + SIG *.36)* CTH          AXIS
      YC = Y + (SIZE / 2.0 -H(NLAB+4) * TNC)*STH                             AXIS
     1+ (-H(NLAB+4) + SIG *(H(NLAB) + H(NLAB+3)))* CTH                       AXIS
C     CALL SYMBOL (XC,YC,0.14,BCD,THETA,NAC)                                 AXIS
      CALL SYMBOL (XC,YC,H(NLAB+2),BCD,THETA,NAC)                            AXIS
      XC = XC +((TNC -6.0) * 0.12)* CTH                                      AXIS
      YC = YC +((TNC -6.0) * 0.12)* STH                                      AXIS
      IF (EXP) 35,50,35                                                      AXIS
C  35 CALL SYMBOL (XC,YC,0.14,@(X10  )@ ,THETA,7)                            AXIS
C  35 CALL SYMBOL (XC,YC,H(NLAB+2),@(X10  )@ ,THETA,7)                       AXIS
   35 CALL SYMBOL (XC,YC,H(NLAB+2),7H(X10  ),THETA,7)                        AXIS
C     XC = XC + .48 * CTH  -.07 * STH                                        AXIS
      XC=XC+.38*CTH-H(NLAB+4)*STH                                            DEC10
C     YC = YC + .48 * STH  +.07 * CTH                                        AXIS
      YC=YC+.38*STH+H(NLAB+4)*CTH                                            DEC10
C  40 CALL NUMBER (XC,YC,0.10,EXP,THETA,-1)                                  AXIS
   40 CALL NUMBER (XC,YC,H(NLAB+3),EXP,THETA,-1)                             AXIS
   50 GO TO 25                                                               AXIS
   60 FNTIC = NTIC                                                           AXIS
      NT = N*NTIC                                                            AXIS
      TN = NT                                                                AXIS
      CTH = CTN/FNTIC                                                        AXIS
      STH = STN/FNTIC                                                        AXIS
      XB = X + TN*CTH                                                        AXIS
      YB = Y + TN*STH                                                        AXIS
      XDELT = - H(6 - NTIC) * SIG  * STH                                     AXIS
      YDELT = H(6 - NTIC) * SIG  * CTH                                       AXIS
      XA = XB + XDELT + XDELT                                                AXIS
C     XA = XB- 0.1 * SIG  * STH                                              AXIS
      YA = YB + YDELT + YDELT                                                AXIS
C     YA = YB+ 0.1 * SIG  * CTH                                              AXIS
      CALL PLOT (XA,YA,3)                                                    AXIS
      XA = XA - XDELT                                                        AXIS
      YA = YA - YDELT                                                        AXIS
C     DO 20  I =1,N                                                          AXIS
      DO 20  I =1,N                                                          AXIS
      DO 20 II =1,NTIC                                                       AXIS
      IF(II.LT.NTIC) GO TO 45                                                AXIS
      XX = XDELT                                                             AXIS
      YY = YDELT                                                             AXIS
      GO TO 46                                                               AXIS
   45 XX = 0.                                                                AXIS
      YY = 0.                                                                AXIS
   46 CONTINUE                                                               AXIS
      CALL PLOT (XB,YB,2)                                                    AXIS
      XC = XB - CTH                                                          AXIS
      YC = YB - STH                                                          AXIS
      CALL PLOT (XC,YC,2)                                                    AXIS
```

```
C       XA = XA - CTH                                                   AXIS
        XA = XA - CTH + XX                                              AXIS
C       YA = YA - STH                                                   AXIS
        YA = YA - STH + YY                                              AXIS
        CALL PLOT (XA,YA,2)                                             AXIS
        XA = XA - XX                                                    AXIS
        YA = YA - YY                                                    AXIS
        XB = XC                                                         AXIS
   20   YB = YC                                                         AXIS
        RETURN                                                          AXIS
        END                                                             AXIS
        SUBROUTINE SCALE (X,S,N,K)                                      SCALE
C                                                                       SCALE
C  WHERE-  X  IS THE NAME OF THE ARRAY OF DATA TO BE SCANNED FOR MAXIMUMSCALE
C             AND MINIMUM VALUES. AN ADJUSTED MINIMUM VALUE WILL BE     SCALE
C             STORED IN X(N*K+1). AN ADJUSTED DX(MAX.-MIN.) WILL BE     SCALE
C             STORED IN X(N*K+K+1).                                     SCALE
C          S  IS THE LENGTH OVER WHICH THIS DATA IS TO BE PLOTTED.      SCALE
C          N  IS THE NUMBER OF DATA POINTS IN THE ARRAY X.              SCALE
C          K  IS THE REPEAT CYCLE OF A MIXED ARRAY.(NORMALLY 1)         SCALE
C                                                                       SCALE
        DIMENSION X(2)                                                  SCALE
        IT = 13                                                         SCALE
        NP = N * K                                                      SCALE
        L = NP + 1                                                      SCALE
        J = NP +K + 1                                                   SCALE
C       XMAX = X(1)                                                     SCALE
C       X(L) = X(1)                                                     SCALE
        DO 2 I=1,NP,K                                                   SCALE
        IF(X(I)) 2,2,3                                                  SCALE
    2   CONTINUE                                                        SCALE
    3   XMAX = X(I)                                                     SCALE
C       X(L)= X(I)                                                      SCALE
        X(L)= 0.0                                                       SCALE
        DO 10 I = 1,NP,K                                                SCALE
        IF(X(I)) 22,10,23                                               SCALE
   22   ITEST = 2.5 - X(I)                                              SCALE
        IF(ITEST) 10,24,10                                              SCALE
   24   X(L) = 0.0                                                      SCALE
        IT = 0                                                          SCALE
        GO TO 10                                                        SCALE
   23   IF (XMAX-X(I)) 5,6,6                                            SCALE
    5   XMAX = X(I)                                                     SCALE
    6   IF (X(L)-X(I))10,10,7                                           SCALE
    7   X(L) = X(I)                                                     SCALE
   10   CONTINUE                                                        SCALE
        DX = (XMAX - X(L)) /S                                           SCALE
        IF (DX) 31,31,30                                                SCALE
   31   X(J) = 1.0                                                      SCALE
        X(L) = X(L) - 0.5                                               SCALE
        RETURN                                                          SCALE
   30   IDX = ALOG10 (DX)                                               SCALE
        IXMN = X(L) * 10.0 ** (-IDX)                                    SCALE
        IF (X(L))32,33,34                                               SCALE
   32   IXMN = X(L) * 10.0 ** (-IDX) - 0.99                             SCALE
   34   X(L) = IXMN                                                     SCALE
        X(L) = X(L) * 10.0 ** IDX                                       SCALE
   33   DX = ALOG10 ((XMAX-X(L))/S)                                     SCALE
        IDX = DX                                                        SCALE
        XMAX = IDX                                                      SCALE
```

```
      DX=10.0**(DX-XMAX)                                                SCALE
      XMAX = 1.0                                                        SCALE
   41 IF (DX-1.0) 40,20,11                                              SCALE
   40 DX = DX * 10.0                                                    SCALE
      IDX = IDX - 1                                                     SCALE
      GO TO 41                                                          SCALE
   11 XMAX=2.0                                                          SCALE
      IF(DX-2.0) 20,20,12                                               SCALE
   12 XMAX = 4.0                                                        SCALE
      IF (DX-4.0) 20,20,13                                              SCALE
   13 XMAX=5.0                                                          SCALE
      IF(DX-5.0) 20,20,14                                               SCALE
   14 XMAX=8.0                                                          SCALE
      IF (DX-8.0) 20,20,15                                              SCALE
   15 XMAX=10.0                                                         SCALE
   20 X(J) = XMAX * 10.0 ** IDX                                         SCALE
      IF(IT) 49,49,39                                                   SCALE
   39 CONTINUE                                                          SCALE
      IMAX = XMAX + 0.00001                                             SCALE
      IXMN = X(L) * 10.0 ** (-IDX)                                      SCALE
      IMOD = IXMN/IMAX*IMAX                                             SCALE
      X(L) = IMOD                                                       SCALE
      X(L) = X(L) * 10.0 ** IDX                                         SCALE
      GO TO 59                                                          SCALE
   49 CONTINUE                                                          SCALE
      X(L) = 0.0                                                        SCALE
   59 CONTINUE                                                          SCALE
      RETURN                                                            SCALE
      END                                                               SCALE
      SUBROUTINE RET                                                    RET
C                                                                       RET
C MSC ENDOCRINE DATA RETREIVAL PROGRAM                                  RET
C     FIRST LOOK STATISTICS PROGRAM.                                    RET
C                                                                       RET
      COMMON XDATA(100),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP        REDUCE
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)          RET
      COMMON /HITBLK/LHIT(100),MDATE(100),ISAV(16),MTVOL(100)           REDUCE
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),   RET
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA           RET
C     COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,      RET
      COMMON /SISBUF/ JBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,      RET
     +                MFUNC,IFILE,IERR                                  RET
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)           RET
      DIMENSION IBUF(14)                                                RET
      EQUIVALENCE(IBUF,T)                                               RET
      EXTERNAL ERROR                                                    RET
C                                                                       RET
C RA1=SAMPLE DIRECTORY  ***  RA2=URINE FILE  ***  RA3=BLOOD FILE        RET
C                       ***  RA4=DAILY FILE  ***                       RET
C                                                                       RET
C                                                                       RET
C PLACE KEYED FILES IN UPDATE MODE EXCLUDING @RA1@ WHICH IS             RET
C             TO BE USED AS A SEQUENTIAL SEARCH FILE                    RET
C             ASSIGNED TO FORTRAN UNIT NO. 1                            RET
C                                                                       RET
      IF(IFFF) 31,201,140                                               RET
C  31 CALL RANDOM(4,@RA2@,125,5,7,100 ,100 ,ICODE,LOC)                  RET
   31 CONTINUE                                                          RET
      CALL FILEIS(FITRA1,3LLFN,3LRA1,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL, 350,3LMNR, 350,3LWSA, T)                       RET
```

```
      CALL STOREF(FITRA1,3LERL,20)                                      RET
      CALL STOREF(FITRA1,2LEX,  ERROR)                                  RET
      CALL OPENM(FITRA1,3LI-O)                                          R
      CALL REWND(FITRA1)                                                R
      CALL FILEIS(FITRA2,3LLFN,3LRA2,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL,1250,3LMNR,1250,3LWSA, T)                       RET
      CALL STOREF(FITRA2,3LERL,0)                                       RET
      CALL STOREF(FITRA2,2LEX,  ERROR)                                  RET
      CALL OPENM(FITRA2,5LINPUT)                                        RET
      CALL REWND(FITRA2)                                                RET
      CALL FILEIS(FITRA3,3LLFN,3LRA3,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL, 350,3LMNR, 350,3LWSA, T)                       RET
      CALL STOREF(FITRA3,3LERL,0)                                       RET
      CALL STOREF(FITRA3,2LEX,  ERROR)                                  RET
      CALL OPENM(FITRA3,5LINPUT)                                        RET
      CALL REWND(FITRA3)                                                RET
      CALL FILEIS(FITRA4,3LLFN,3LRA4,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL, 300,3LMNR, 300,3LWSA,T)                        RET
      CALL STOREF(FITRA4,3LERL,0)                                       RET
      CALL STOREF(FITRA4,2LEX,  ERROR)                                  RET
      CALL OPENM(FITRA4,5LINPUT)                                        RET
      CALL REWND(FITRA4)                                                RET
      IFFF = 0                                                          RET
C                                                                       RET
C INITIALIZE COUNTER FLAGS FOR THE FIRST PASS OF                        RET
C    ANY GIVEN RETREIVAL                                                RET
C                                                                       RET
  201 ISFLG = 1                                                         RET
      IDFLG = 1                                                         R
      IMFLG = 1                                                         R
      ITPFLG = 1                                                        RET
      ITTFLG = 1                                                        RET
      IFFF = 1                                                          RET
C                                                                       RET
C READ ONE DATA CARD                                                    RET
C                                                                       RET
    1 READ 200,(ISAV(I),I=1,16)                                         RET
  200 FORMAT(A4,1X,A4,1X,14I5)                                          RET
C                                                                       RET
C DETERMINE CARD TYPE AND COUNTER SEQUENCE                              RET
C                                                                       RET
      IF(ISAV(1).EQ.4HJOB ) GO TO 19                                    RET
      IF(ISAV(1).EQ.4HSTUD.AND.ISFLG.EQ.1) GO TO  5                     RET
      IF(ISAV(1).EQ.4HSTUD.AND.ISFLG.EQ.0) GO TO 45                     RET
      IF(ISAV(1).EQ.4HDATE.AND.IDFLG.EQ.1) GO TO 6                      RET
      IF(ISAV(1).EQ.4HDATE.AND.IDFLG.EQ.0) GO TO 46                     RET
      IF(ISAV(1).EQ.4HMAN .AND.IMFLG.EQ.1) GO TO  7                     RET
      IF(ISAV(1).EQ.4HMAN .AND.IMFLG.EQ.0) GO TO 47                     RET
      IF(ISAV(1).EQ.4HTYPE.AND.ITPFLG.EQ.1) GO TO  8                    RET
      IF(ISAV(1).EQ.4HTYPE.AND.ITPFLG.EQ.0) GO TO 48                    RET
      IF(ISAV(1).EQ.4HTEST.AND.ITTFLG.EQ.1) GO TO  9                    RET
      IF(ISAV(1).EQ.4HTEST.AND.ITTFLG.EQ.0) GO TO 49                    RET
      IF(ISAV(1).EQ.4HEND ) GO TO 40                                    RET
      IF(ISAV(1).EQ.4HEOF ) GO TO 99                                    RET
C                                                                       RET
C BUILD RETREIVAL CRITERIA ARRAYS                                       R
C                                                                       RET
   5  JSTALL=0                                                          RET
      IS = 0                                                            RET
      ISFLG = 0                                                         RET
```

```
   45 CALL SETCRI(LSTUDY,IS,JSTALL)                                    RET
C     OUTPUT,(LSTUDY(I),I=1,5),IS,JSTALL                               RET
      GO TO 1                                                          RET
    6 JDTALL=0                                                         RET
      ID = 0                                                           RET
      IDFLG = 0                                                        RET
   46 CALL SETCRI(LDATE,ID,JDTALL)                                     RET
C     OUTPUT,(LDATE(I),I=1,10),ID,JDTALL                               RET
      GO TO 1                                                          RET
    7 MANALL=0                                                         RET
      IM = 0                                                           RET
      IMFLG = 0                                                        RET
   47 CALL SETCRI(LMAN,IM,MANALL)                                      RET
C     OUTPUT,(LMAN(I),I=1,9),IM,MANALL                                 RET
      GO TO 1                                                          RET
    8 JTPALL=0                                                         RET
      ITP = 0                                                          RET
      ITPFLG = 0                                                       RET
   48 CALL SETCRI(LTYPE,ITP,JTPALL)                                    RET
C     OUTPUT,(LTYPE(I),I=1,10),ITP,JTPALL                              RET
      GO TO 1                                                          RET
    9 JTTALL=0                                                         RET
      ITT = 0                                                          RET
      ITTFLG = 0                                                       RET
   49 CALL SETCRI(LTEST,ITT,JTTALL)                                    RET
      ITT=1                                                            RET
C     IF(JTTALL.EQ.1) ITT = 124                                        RET
      LSTART = ISAV(3)                                                 RET
      LSTOP  = ISAV(4)                                                 RET
      IF(LSTOP.EQ.0) LSTOP=LSTART                                      DEC10
      GO TO 1                                                          RET
   19 DO 119 I=1,16                                                    RET
  119 JOB(I)=ISAV(I)                                                   RET
      IF(IOPLOT.EQ.0.AND.JOB(3).GE.1000) CALL PLOTS(14HNASA ENDOCRINE,  RET
     1  14)                                                            RET
      IF(JOB(3).GE.1000) IOPLOT = 999                                  RET
      IF(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH.OR.JOB(2).EQ.4HVOLU)GOTO1 DEC10
      JOB(2)=4HDATE                                                    DEC10
      PRINT 601                                                        DEC10
  601 FORMAT(* UNKNOWN JOB PARAMETER DATE ASSUMED*)                    DEC10
      GO TO 1                                                          RET
C                                                                      RET
C BUILD HIT ARRAY OF SAMPLE NUMBERS                                    RET
C                                                                      RET
   40 CONTINUE                                                         RET
      LT = LSTART - 1                                                  RET
C                                                                      RET
C     DO 4321 LT=LSTART,LSTOP                                          RET
  140 IF(IFFF - 2) 145,240,145                                         RET
  145 LT = LT + 1                                                      RET
C                                                                      RET
      IF(LT - LSTOP) 150,150,4321                                      RET
  150 LM = 0                                                           RET
      IFFF = 2                                                         RET
C     IF(KOT.EQ.0) CALL SSWTCH(1,KK1)                                  RET
C     IF(KKT.EQ.1) GO TO 999                                           RET
      LTEST(1) = LT                                                    RET
C     DO 4321 LM=1,3                                                   RET
  240 LM = LM + 1                                                      RET
      IF(LM - 3) 250,250,245                                           RET
```

```
  245 IFFF = 1                                                          RET
      GO TO 140                                                         RET
  250 LMAN(1) = LM                                                      R
      IH = 1                                                            R
C     REWIND 1                                                          RET
      CALL REWND(FITRA1)                                                RET
C                                                                       RET
C                                                                       RET
C THE KEYED (RA1) SAMPLE DIRECTORY FILE IS TREATED AS A SEQUENTIAL      RET
C     FILE DURING THE SEARCH FOR SAMPLES WHICH SATISFY                  RET
C                    THE INPUT CRITERIA                                 RET
C                                                                       RET
C   10 BUFFER IN(1,1)(IBUF(1),IBUF(14))                                 RET
   10 CALL GETN(FITRA1)                                                 RET
      IF(IFETCH(FITRA1,2LFP).EQ.100B) GO TO 23                          RET
    2 IF(JSTALL.EQ.1) GO TO 12                                          RET
      DO 11 I=1,IS                                                      RET
      IF(IBUF(2).EQ.LSTUDY(I)) GO TO 12                                 RET
   11 CONTINUE                                                          RET
      GO TO 10                                                          RET
   12 IF(JDTALL.EQ.1.AND.JOB(2).EQ.4HVOLU) GO TO 44                     RET
C                                                                       RET
C CHECK FOR SIMULTANEOUS @ALL@ AND DATE @SAVE@ OPTION                   RET
C                                                                       RET
      IF(JDTALL.EQ.1.AND.(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH))GO TO 41RET
      DO 13 I=1,ID                                                      RET
      IF(IBUF(3).EQ.LDATE(I)) GO TO 14                                  RET
   13 CONTINUE                                                          RET
      GO TO 10                                                          RET
   41 MDSAV = IBUF(3)                                                   R
      GO TO 44                                                          RET
C                                                                       RET
C CHECK FOR DATE SAVE OPTION                                            RET
C                                                                       RET
   14 IF(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH) MDSAV=LDATE(I)           RET
   44  IF(MANALL.EQ.1) GO TO 16                                         RET
      DO 15 I=1,IM                                                      RET
      IF(IBUF(4).EQ.LMAN(I)) GO TO 16                                   RET
   15 CONTINUE                                                          RET
      GO TO 10                                                          RET
   16 IF(JTPALL.EQ.1) GO TO 18                                          RET
      DO 17 I=1,ITP                                                     RET
      IF(IBUF(5).EQ.LTYPE(I)) GO TO 18                                  RET
   17 CONTINUE                                                          RET
      GO TO 10                                                          RET
C                                                                       RET
C CHECK FOR MASTER SAMPLE NUMBER                                        RET
C                                                                       RET
   18 IF(IBUF(14).EQ.0) GO TO 21                                        RET
      MIH = IH-1                                                        RET
      DO 20 I=1,MIH                                                     RET
      IF(LHIT(I).EQ.IBUF(14)) GO TO 10                                  RET
   20 CONTINUE                                                          RET
      LHIT(IH)=IBUF(14)                                                 RET
C     PRINT 220,(LHIT(I),I=1,10)                                        RET
C 220 FORMAT(5X,5HLHIT2,5X,10I10)                                       R
      GO TO 22                                                          RET
   21 LHIT(IH)=IBUF(1)                                                  RET
C     PRINT 221,(LHIT(I),I=1,10)                                        RET
C 221 FORMAT(5X,5HLHIT1,5X,10I10)                                       RET
```

```
C                                                                       RET
C  IF DATE SAVE OPTION WAS SPECIFIED FILL DATE ARRAY CORRESPONDING      RET
C     TO SAMPLE NUMBER HIT ARRAY                                        RET
C                                                                       RET
   22 IF(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH) MDATE(IH)=MDSAV          RET
C                                                                       RET
C  IF TOTAL VOLUME SAVE OPTION WAS SPECIFIED FILL VOLUME ARRAY          RET
C         CORRESPONDING TO SAMPLE NUMBER HIT ARRAY                      RET
C                                                                       RET
      IF(JOB(2).EQ.4HVOLU.OR.JOB(2).EQ.4HBOTH)MTVOL(IH)=IBUF(10)        RET
       IH=IH+1                                                          RET
      IF(IH.LE.101) GO TO 10                                            REDUCE
   23 NSMP = IH - 1                                                     RET
      IF(NSMP.LE.100) GO TO 35                                          REDUCE
      PRINT 30                                                          RET
   30 FORMAT(1H1,17HHIT FILE OVERFLOW    )                              RET
      STOP 30                                                           RET
C                                                                       RET
   35 CALL RETRVE                                                       RET
      PRINT 300, LSTUDY,LMAN,JOB,LTYPE,NSMP,NDATA                       RET
  300 FORMAT(1H0,6HLSTUDY,10I10,//,2X,4HLMAN,9I10,//,                   DEC10
     .   1X, 4HJOB ,A4,5X,A4,5X,14I6,//,6H LTYPE,20I5,//,              RET
     .   1X, 7HNSMP = ,I5,10X,6HNDATA = ,I5)                           DEC10
      PRINT 301,(LDATE(I),I=1,ID)                                       RET
  301 FORMAT(6H LDATE /,(10I10))                                        RET
      PRINT 302,(LTEST(I),I=1,ITT)                                      RET
  302 FORMAT(6H LTEST  ,(10I10))                                        RET
      PRINT 306,(MTVOL(I),I=1,NSMP)                                     RET
  306 FORMAT(1H0,5HMTVOL,/,(10I10))                                     DEC10
      PRINT 303,(MDATE(I),I=1,NSMP)                                     RET
  303 FORMAT(1H0,5HMDATE,/,(10I10))                                     DEC10
      PRINT 304,(LHIT(I),I=1,NSMP)                                      RET
  304 FORMAT(/ ,6H LHIT ,/,(10I10))                                     DEC10
      PRINT 305,(XDATA(I),I=1,NDATA)                                    RET
  305 FORMAT(1H0,5HXDATA,/,(8F15.6))                                    DEC10
C     PRINT 300, LSTUDY,LMAN,JOB,LTYPE,NSMP,NDATA                       RET
C 300 FORMAT(1H1,6HLSTUDY,10I10,//,2X,4HLMAN,9I10,//,                   RET
C    .       4HJOB ,A4,5X,A4,5X,14I6,//,5HLTYPE,20I5,//,                RET
C    .       7HNSMP = ,I5,10X,8HNDATA = ,I5,// )                        RET
C     PRINT 301,(LDATE(I),I=1,ID)                                       RET
C 301 FORMAT(5HLDATE /,(10I10))                                         RET
C     PRINT 302,(LTEST(I),I=1,ITT)                                      RET
C 302 FORMAT(5HLTEST  ,(10I10))                                         RET
C     PRINT 306,(MTVOL(I),I=1,NSMP)                                     RET
C 306 FORMAT(1H1,5HMTVOL,/,(10I10))                                     RET
C     PRINT 303,(MDATE(I),I=1,NSMP)                                     RET
C 303 FORMAT(1H1,5HMDATE,/,(10I10))                                     RET
C     PRINT 304,(LHIT(I),I=1,NSMP)                                      RET
C 304 FORMAT(//,5HLHIT ,/,(10I10))                                      RET
C     PRINT 305,(XDATA(I),I=1,NDATA)                                    RET
C 305 FORMAT(1H1,5HXDATA,/,(8F15.6))                                    RET
C                                                                       RET
C                                                                       RET
C     ------------------------------------------------------------------RET
      RETURN                                                            RET
C     CALL STAT                                                         RET
 4321 CONTINUE                                                          RET
      IFFF = 0                                                          RET
C                                                                       RET
C     ------------------------------------------------------------------RET
```

```
C                                                                       RET
C     REWIND 1                                                          RET
      CALL REWND(FITRA1)                                                RET
      GO TO 201                                                         RE
C                                                                       RE
  888 PRINT 8800                                                        RET
 8800 FORMAT(15H FILE 1 REWOUND)                                        RET
      RETURN                                                            RET
   99 CONTINUE                                                          RET
      CALL CLOSEM(FITRA1)                                               RET
      CALL CLOSEM(FITRA2)                                               RET
      CALL CLOSEM(FITRA3)                                               RET
      CALL CLOSEM(FITRA4)                                               RET
      IF(IOPLOT) 199,199,999                                            RET
  999 CALL STOPPLT                                                      RET
      PRINT 900                                                         RET
  900 FORMAT(19HONORMAL END OF JOB.)                                    RET
      STOP 2000                                                         RET
  199 STOP 1                                                            RET
      END                                                               RET
      SUBROUTINE SETCRI(LBUF,JNDX,LALL)                                 SETCRI
C  THIS SUBROUTINE STORES INFORMATION FROM ONE DATA CARD INTO THE        SETCRI
C                          CORRESPONDING CRITERIA ARRAY                 SETCRI
C                                                                       SETCRI
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)          SETCRI
      COMMON /HITBLK/LHIT(100),MDATE(100),ISAV(16),MTVOL(100)           REDUCE
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),   SETCRI
     .                LTYPE(20),ITP,LIEST(125),ITT,NSMP,NDATA           SETCRI
      COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,      SETCRI
     +                MFUNC,IFILE,IERR                                  SE  I
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)           SE  I
      DIMENSION LBUF(1)                                                 SETCRI
      IF(ISAV(2).EQ.4HALL .OR.ISAV(2).EQ.4H    ) GO TO 1                SETCRI
      JNDX = JNDX + 1                                                   SETCRI
      LBUF(JNDX)=ISAV(3)                                                SETCRI
      IF(ISAV(2).EQ.4HEACH) GO TO 3                                     SETCRI
      INXT=ISAV(3)+1                                                    SETCRI
    2 IF(INXT.GT.ISAV(4)) RETURN                                        SETCRI
      JNDX = JNDX + 1                                                   SETCRI
      LBUF(JNDX)=INXT                                                   SETCRI
      INXT=INXT+1                                                       SETCRI
      GO TO 2                                                           SETCRI
    3 DO 5 I=4,16                                                       SETCRI
      IF(ISAV(I).EQ.0) RETURN                                           SETCRI
      JNDX = JNDX + 1                                                   SETCRI
      LBUF(JNDX)=ISAV(I)                                                SETCRI
    5 CONTINUE                                                          SETCRI
      RETURN                                                            SETCRI
    1 LALL=1                                                            SETCRI
      RETURN                                                            SETCRI
      END                                                               SETCRI
      SUBROUTINE RETRVE                                                 RETRVE
C                                                                       RETRVE
C THIS SUBROUTINE RETREVES THOSE DATA VALUES FROM THE BLOOD OR          RETRVE
C      URINE FILES WHICH SATISFY THE INPUT CRITERIA                     RETRVE
C      THE DATA VALUES ARE STORED IN ARRAY XDATA                        RE  VE
C                                                                       RE  VE
      COMMON XDATA(100),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP        REDUCE
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)          RETRVE
      COMMON/INDEX/INDX(136),INDX3(136)                                 DEC10
```

```
      COMMON /HITBLK/LHIT(100),MDATE(100),ISAV(16),MTVOL(100)          REDUCE
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),  RETRVE
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA          RETRVE
      COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEE,KYE,     RETRVE
     +                MFUNC,IFILE,IERR                                 RETRVE
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)          RETRVE
      DIMENSION XTEMP(125)                                             RETRVE
      INTEGER KEY(2)                                                   RETRVE
      EQUIVALENCE(T,XTEMP)                                             RETRVE
      EQUIVALENCE (KEE,KEY(2))                                         RETRVE
      DATA(INDX(I),I=1,136)/400,401,402,403,404,405,503,406,407,408,409,DEC10
     .  410,                                                           DEC10
     1  411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,426,DEC10
     2  427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,DEC10
     3  443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,DEC10
     4  459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,DEC10
     5  475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,490,DEC10
     6  491,492,493,494,495,496,497,498,499,500,501,502,300,301,302,303,DEC10
     7  304,305,306,307,321,308,309,310,311,312,313,314,315,316,317,318,DEC10
     .  319,                                                           DEC10
     8  320,514,515,516,517,518,519,520,521,522,523/                  DEC10
      DATA (INDX3(I),I=1,136)/2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,  DEC10
     1  18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,  DEC10
     2  39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,  DEC10
     3  60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,  DEC10
     4  81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,    DEC10
     5101,102,103,104,105,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,DEC10
     6  20,21,22,23,116,117,118,119,120,121,122,123,124,125/           DEC10
      DATA ICDT/445B/                                                  RETRVE
      M=1                                                              RETRVE
      KSAV=0                                                           RETRVE
      IERR=0                                                           RETRVE
      DO 27 K=1,NSMP                                                   RETRVE
      DO 27 I=1,ITT                                                    RETRVE
C                                                                      RETRVE
C   IDENTIFY TEST NUMBER IN INDEX                                      RETRVE
C                                                                      RETRVE
      DO 24 J=1,136                                                    DEC10
      IF(LTEST(I).EQ.INDX(J)) GO TO 25                                 DEC10
   24 CONTINUE                                                         DEC10
      STOP @INVALID TEST NUMBER@                                       DEC10
   25 IJ=J                                                             DEC10
      LREC=INDX3(IJ)                                                   DEC10
      IF(LTEST(I) -400) 41,31,31                                       RETRVE
   31 INDX1 = 1RU                                                      RETRVE
      GO TO 50                                                         RETRVE
   41 INDX1 = 1RB                                                      RETRVE
C                                                                      RETRVE
C   PICK UP KEY FROM HIT ARRAY OF SAMPLE NUMBERS                       RETRVE
C                                                                      RETRVE
   50 KEY(2)=LHIT(K)                                                   RETRVE
C                                                                      RETRVE
C  DETERMINE IF DATA NEEDED IS FROM CURRENT RECORD                     RETRVE
C                                                                      RETRVE
      IF(KEY(2).EQ.KSAV) GO TO 26                                      RETRVE
C                                                                      RETRVE
C   CHECK FOR URINE OR BLOOD FILE                                      RETRVE
C                                                                      RETRVE
C  29 IF(INDX1(IJ).EQ.1RU) GO TO 28                                    RETRVE
   29 IF(INDX1.EQ.1RU) GO TO 28                                        RETRVE
```

```
C      CALL RANDOM(1,@RA3@,XTEMP,32,KEY,100 ,100 ,ICODE,LOC)                RETRVE
       CALL GET(FITRA3)                                                     RETRVE
       LOC = 3                                                              RETRVE
       IFILE = 3                                                            RETRVE
       IF(IERR.NE.0) GO TO 100                                              RETRVE
       GO TO 26                                                             RETRVE
C   28 CALL RANDOM(1,@RA2@,XTEMP,125,KEY,100 ,100 ,ICODE,LOC)               RETRVE
    28 CALL GET(FITRA2)                                                     RETRVE
       LOC = 2                                                              RETRVE
       IFILE = 2                                                            RETRVE
       IF(IERR.NE.0) GO TO 100                                              RETRVE
C                                                                           RETRVE
C                                                                           RETRVE
C FETCH WORD NUMBER OF DATA RECORD FROM INDEX, STORE DATA                   RETRVE
C    VALUE IN CURRENT XDATA LOCATION AND INCREMENT                          RETRVE
C           XDATA LOCATION COUNTER M                                        RETRVE
C                                                                           RETRVE
C                                                                           RETRVE
C   26 LREC=INDX3(IJ)                                                       RETRVE
    26 XDATA(M)=XTEMP(LREC)                                                 RETRVE
    30 M=M+1                                                                RETRVE
       IF(M.GT.100) GO TO 110                                               REDUCE
       KSAV=KEY(2)                                                          RETRVE
    27 CONTINUE                                                             RETRVE
C                                                                           RETRVE
C                                                                           RETRVE
C NDATA = NUMBER OF DATA VALUES RETREIVED                                   RETRVE
C                                                                           RETRVE
C                                                                           RETRVE
       NDATA=M-1                                                            RETRVE
       RETURN                                                               RETRVE
  100 IF(IERR .EQ.ICDT) GO TO 105                                           RETRVE
       PRINT 101,IERR ,KEY(2)                                               RETRVE
  101 FORMAT(1H ,19HRANDOM ERROR-RETRVE,08,5X,I4)                           RETRVE
       STOP 100                                                             RETRVE
  105 PRINT 104,KEY(2)                                                      RETRVE
  104 FORMAT(1H ,27HSAMPLE NO. NOT FOUND-RETRVE,5X,I4)                      RETRVE
       XDATA(M)=-1.                                                         RETRVE
       IERR=0                                                               RETRVE
       GO TO 30                                                             RETRVE
  110 PRINT 111                                                             RETRVE
  111 FORMAT(1H1,43HDATA BUFFER FULL BEFORE ALL DATA RETREIVED   )          RETRVE
       STOP 110                                                             RETRVE
       END                                                                  RETRVE
       SUBROUTINE STAT                                                      STAT
       COMMON XDATA(100),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP           REDUCE
       COMMON /HITBLK/LHIT(100),MDATE(100),ISAV(16),MTVOL(100)              REDUCE
       COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),      STAT
      .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA              STAT
       COMMON/RG/ DATESC(3,5),IVV(4), SSMEAN(3), SSIGMA(3), NNN(3),SSE(3)   STAT
       REAL KSTUDY, KTYPE, KMAN, KTEST, NITS                                STAT
       DIMENSION ISIG(2)                                                    STAT
       DIMENSION X(1)                                                       STAT
       DIMENSION LTEXT(120), MTEXT(120), XOUT(120)                          STAT
       DIMENSION NDF(3), S(3), P(3), TM(3), WTV(3), XMS(3), IPP(3),IP2(3)   STAT
       DIMENSION ID1(120), ID2(120), LK(3)                                  ST
       EQUIVALENCE(X,XDATA)                                                 ST
       DATA ID1/120*1/, LK/3*0/                                             STAT
       DATA IPP/4HPRE , 4HIN  , 4HPOST/                                     STAT
       DATA IP2/8HPRE-IN  , 8HIN-POST , 8HPRE-POST/                         STAT
```

```
C                                                                       STAT
C     FILL OUTPUT ARRAY                                                 STAT
C                                                                       STAT
      DATA ISTAR/1H*/, IBLANK/1H /, ISAVE/4HSAVE/                       STAT
      DATA IPLUS/1H+/, IMINUS/1H-/, IPPL/2H++/, IMMI/2H--/              STAT
      DATA IDATE/4HDATE/, IVOLU/4HVOLU/, IBOTH/4HBOTH/                  STAT
      NOBS = NDATA                                                      STAT
    1 IF(NOBS - 1) 999,999,2                                            STAT
    2 LOOP = ITT                                                        STAT
      IDATEF = 0                                                        STAT
      IVOLUF = 0                                                        STAT
      IF(JOB(2).EQ.IDATE) IDATEF = 3                                    STAT
      IF(JOB(2).EQ.IVOLU) IVOLUF = 3                                    STAT
      IF(JOB(2).NE.IBOTH) GO TO 3                                       STAT
      IDATEF = 3                                                        STAT
      IVOLUF = 3                                                        STAT
    3 KKK = NOBS/LOOP                                                   STAT
      CALL TEXT(LSTUDY(1), LTYPE(1), LMAN(1), LTEST(1),                 STAT
     X KSTUDY, KTYPE, KMAN, KTEST, NITS)                                STAT
      CALL DATE(IWORD)                                                  STAT
      PRINT 1000, JOB(3)                                                STAT
 1000 FORMAT(1H1,20X,42HNASA MSC ENDOCRINE DATA RETREIVAL PROGRAM.,I7/) STAT
      PRINT 500, LSTUDY(1), KSTUDY,IWORD                                STAT
  500 FORMAT(7H0STUDY,,I2,2X, A8,20X,A10)                               STAT
      PRINT 600, LTEST(1), KTEST,LMAN(1), KMAN                          STAT
  600 FORMAT(6HJTEST,,I3,2X A8 ,20X,4HMAN,,I4,2X, A8)                   STAT
      PRINT 700,(LTYPE(I),I=1,ITP)                                      DEC10
  700 FORMAT(6H0TYPE,,20I3)                                             DEC10
      PRINT 800, NITS                                                   STAT
  800 FORMAT(15 10JULIAN DATE     , A3)                                 STAT
C     DO 150 L=1,LOOP                                                   STAT
      L = 1                                                             STAT
      LOOP = 1                                                          STAT
      DO 210 II=LOOP,NOBS,LOOP                                          STAT
      I = II + L - LOOP                                                 STAT
      IPLACE = II/LOOP                                                  STAT
  210 XOUT(IPLACE) = XDATA(I)                                           STAT
C                                                                       STAT
C     SORT BY DATE                                                      STAT
C                                                                       STAT
      NOSLTS = NOBS/LOOP                                                STAT
      NM1 = NOSLTS - 1                                                  STAT
      DO 230 I=1,NM1                                                    STAT
      IMAX = 9999                                                       STAT
      ISUB = I                                                          STAT
      DO 220 J=I,NOSLTS                                                 STAT
      IF(IMAX-MDATE(J)) 220,220,211                                     STAT
  211 IMAX = MDATE(J)                                                   STAT
      ISUB = J                                                          STAT
  220 CONTINUE                                                          STAT
      ITEMP = MDATE(ISUB)                                               STAT
      TEMP = XOUT(ISUB)                                                 STAT
      MDATE(ISUB) = MDATE(I)                                            STAT
      XOUT(ISUB) = XOUT(I)                                              STAT
      MDATE(I) = ITEMP                                                  STAT
      XOUT(I) = TEMP                                                    STAT
  230 CONTINUE                                                          STAT
      IG = LSTUDY(1)                                                    STAT
      IVV(1) = 0                                                        STAT
      IVV(4) = NOSLTS                                                   STAT
```

```
      DO 240 I=1,NOSLTS                                                  STAT
      IF(MDATE(I).LE.DATESC(2,IG)) IVV(2) = I                            STAT
      IF(MDATE(I).LE.DATESC(3,IG)) IVV(3) = I                            STAT
  240 XDATA(I) = XOUT(I)                                                 STAT
      DO 133 I=1,3                                                       STAT
      NFIRST = IVV(I) + 1                                                STAT
      NLAST = IVV(I + 1)                                                 STAT
      CALL MESIG(LTEXT,NFIRST,NLAST,SMEAN,SIGMA,NN)                      STAT
      SSMEAN(I) = SMEAN                                                  STAT
      SSIGMA(I) = SIGMA                                                  STAT
      NNN(I) = NN                                                        STAT
  133 CONTINUE                                                           STAT
      NFIRST = 1                                                         STAT
      NLAST = NOSLTS                                                     STAT
      CALL MESIG(LTEXT,NFIRST,NLAST,SMEAN,SIGMA,NN)                      STAT
C                                                                        STAT
C     IF THERE ARE 2 OR MORE PRE-FLIGHT OBSERVATIONS THE EXTREME VALUES  STAT
C     ARE FLAGED ON THE BASIS OF PRE-FLIGHT MEAN AND STD. DEV.           STAT
C     OTHERWISE THE TOTAL MEAN AND STD. DEV. ARE USED.                   STAT
C                                                                        STAT
      IF(NNN(1) - 2) 134,136,136                                         STAT
  134 SSG = SIGMA                                                        STAT
      SSM = SMEAN                                                        STAT
      GO TO 137                                                          STAT
  136 SSG = SSIGMA(1)                                                    STAT
      SSM = SSMEAN(1)                                                    STAT
  137 T01 = 2.3263*SSG                                                   STAT
      T05 = 1.6449*SSG                                                   STAT
      DO 166 II=1,3                                                      STAT
      IFIRST = IVV(II)+ 1                                                STAT
      ILAST = IVV(II + 1)                                                STAT
      IF(IFIRST - ILAST) 162,166,166                                     STAT
  162 DO 165 I=IFIRST,ILAST                                              STAT
      MTEXT(I) = IBLANK                                                  STAT
      IF(X(I)) 164,165,153                                               STAT
  153 DELTA = X(I) - SSM                                                 STAT
      IF(DELTA) 155,154,157                                              STAT
  154 IF(X(I).NE.2.0) GO TO 164                                          STAT
      DELTA = SSM                                                        STAT
      GO TO 156                                                          STAT
  155 DELTA = -DELTA                                                     STAT
  156 IF(DELTA.GE.T05) MTEXT(I) = IMINUS                                 STAT
      IF(DELTA.GE.T01) MTEXT(I) = IMMI                                   STAT
      GO TO 164                                                          STAT
  157 IF(DELTA.GE.T05) MTEXT(I) = IPLUS                                  STAT
      IF(DELTA.GE.T01) MTEXT(I) = IPPL                                   STAT
  164 PRINT 400,MDATE(I), X(I), LTEXT(I), MTEXT(I)                       STAT
  165 CONTINUE                                                           STAT
      PRINT 900,SSMEAN(II), SSIGMA(II), NNN(II)                          STAT
  166 CONTINUE                                                           STAT
      IF(JOB(3) - 1000) 150,145,145                                      STAT
  145 IF(NN) 150,150,146                                                 STAT
  146 CALL STPLOT(LOOP,NOBS,NSLOT,IWORD,KSTUDY,KTYPE,KMAN,KTEST,SMEAN,   STAT
     X SIGMA)                                                            STAT
  150 CONTINUE                                                           STAT
C                                                                        STAT
C     ICASE = 0     NO TESTS                                           STAT
C     ICASE = 1     PREFLIGHT ONLY                                     STAT
C     ICASE = 2     PRE AND INFLIGHT ONLY                              STAT
C     ICASE = 3     ALL                                                STAT
```

```
C      ICASE = 4     PRE AND POSTFLIGHT ONLY                                  STAT
C                                                                             STAT
       IF(NNN(1)-1) 170,170,171                                               DEC10
  170  NT = 0                                                                 STAT
       ICASE = 0                                                              STAT
       GO TO 999                                                              DEC10
  171  SSE(1) = SSIGMA(1)/SQRT(FLOAT(NNN(1)))                                 STAT
       PRINT 3000,NNN(1), SSMEAN(1), SSIGMA(1), SSE(1)                        STAT
 3000  FORMAT(20(4H----)/23H PARAMETRIC STATISTICS.//8H0SUMMARY/              STAT
     X39H0SAMPLE  N     MEAN          SD          SE/4H0PKT, I6, 3F11.3)     STAT
       IF(NNN(2)-1) 172,172,176                                               DEC10
  172  IF(NNN(3)-1) 173,173,174                                               DEC10
  173  NT = 1                                                                 STAT
       ICASE = 1                                                              STAT
       GO TO 999                                                              DEC10
  174  SSE(3) = SSIGMA(3)/SQRT(FLOAT(NNN(3)))                                 STAT
       PRINT3010,NNN(3), SSMEAN(3), SSIGMA(3), SSE(3)                         STAT
       NT = 2                                                                 STAT
       ICASE = 4                                                              STAT
C                                                                             STAT
C      FOR CASE 4 FIX NNN AND X FOR ACRDAN                                    DEC10
       IF(NNN(2).EQ.1) GO TO 175                                              DEC10
       NNN(2)=NNN(3)                                                          DEC10
       GO TO 180                                                              DEC10
C      NNN(2) IS 1 MOVE X                                                     DEC10
  175  CONTINUE                                                               DEC10
       NNN1=NNN(1)+1                                                          DEC10
       NSLTM1=NOSLTS-1                                                        DEC10
       DO 179 I=NNN1,NSLTM1                                                   DEC10
       X(I)=X(I+1)                                                            DEC10
  179  CONTINUE                                                               DEC10
       NNN(2)=NNN(3)                                                          DEC10
       GO TO 180                                                              DEC10
C      CHANGE THIS LATER                                                      STAT
C                                                                             STAT
  176  SSE(2) = SSIGMA(2)/SQRT(FLOAT(NNN(2)))                                 STAT
       PRINT3020,NNN(2), SSMEAN(2), SSIGMA(2), SSE(2)                         STAT
 3020  FORMAT(4H0IM ,I6,3F11.3)                                               STAT
       IF(NNN(3)-1) 177,177,178                                               DEC10
  177  NT = 2                                                                 STAT
       ICASE = 2                                                              STAT
       GO TO 180                                                              STAT
  178  SSE(3) = SSIGMA(3)/SQRT(FLOAT(NNN(3)))                                 STAT
       PRINT3010,NNN(3), SSMEAN(3), SSIGMA(3), SSE(3)                         STAT
 3010  FORMAT(5H0POST,I5,3F11.3)                                              STAT
       NT = 3                                                                 STAT
       ICASE = 3                                                              STAT
  180  CONTINUE                                                               STAT
       SE = SIGMA/SQRT(FLOAT(NN))                                             STAT
       PRINT 3030, NN, SMEAN, SIGMA, SE                                       STAT
 3030  FORMAT(4H ---,10(4H----)/4H SUM,I6,3F11.3)                              STAT
C                                                                             STAT
C      GET RID OF NEGATIVE VALUES                                             STAT
C         I  RANGES OVER ALL POINTS.                                          STAT
C         J  RANGES OVER VALID POINTS.                                        STAT
C                                                                             STAT
       J = 1                                                                  STAT
       DO 330 I=1,NOSLTS                                                      STAT
       IF(X(I)) 310,330,320                                                   STAT
  310  IF(X(I) + 2.0) 330,315,330                                             STAT
```

```
 315 X(J) = 0.0                                                          STAT
     J = J + 1                                                           STAT
     GO TO 330                                                           STAT
 320 X(J) = X(I)                                                         STAT
     J = J + 1                                                           STAT
 330 CONTINUE                                                            STAT
     CALL ACRDAN(X, NT, NNN, TM, WTV, S, GM, NDF, IER)                   STAT
     N1 = NDF(1)                                                         STAT
     N2 = NDF(2)                                                         STAT
     S1 = S(1)                                                           STAT
     S2 = S(2)                                                           STAT
     CALL FFOUT(S1,S2,N1,N2,FVAL,PRFVAL)                                 STAT
     DO 193 IMS=1,3                                                      STAT
 193 XMS(IMS) = S(IMS)/NDF(IMS)                                          STAT
     PRINT 3090                                                          STAT
3090 FORMAT(1X)                                                          STAT
     PRINT 3100                                                          STAT
3100 FORMAT(28H0ANALYSIS OF VARIANCE TABLE,/35H0        DF       SS      STAT
    X  MS      F /35H        ----     ----       ----    ---)           STAT
     PRINT 3200, NDF(1), S(1), XMS(1), FVAL                              STAT
3200 FORMAT(6H TREAT, I4, F10.1, F10.1, F8.3)                            STAT
     PRINT 3300, NDF(2), S(2), XMS(2)                                    STAT
3300 FORMAT(6H0ERROR, I4, F10.1, F10.1, 6H  --- /4H ---, 9(4H----))      STAT
     PRINT 3400, NDF(3), S(3), XMS(3)                                    STAT
3400 FORMAT(6H TOTAL, I4, F10.1, F10.1, 6H  --- )                        STAT
     PRINT 3800, FVAL, PRFVAL                                            STAT
3800 FORMAT(1H0,F7.3, 23H IS SIGNIFICANT AT THE  ,F5.1, 14H PERCENT LEVSTAT
    1EL)                                                                 STAT
     IF(NT-2) 999,850,824                                                DEC10
 824 IF(PRFVAL - 5.0) 825,825,850                                        STAT
 825 N1 = 1                                                              S
     PRINT 3500                                                          STAT
3500 FORMAT(14H0CONTRAST   F )                                           STAT
     P(1) = 1                                                            STAT
     P(2) = -1                                                           STAT
     CALL ACTRST(TM,  NNN, 3, 1, P, 1, Q, SQ)                            STAT
     CALL FFOUT(SQ,S2,N1,N2,FVAL,PRFVAL)                                 STAT
     PRINT 3600,IP2(1), IP2(2), FVAL, PRFVAL                             STAT
3600 FORMAT(1H0, 2A4, F8.3, 23H IS SIGNIFICANT AT THE  ,F5.1,            STAT
    X15H PERCENT LEVEL.)                                                 STAT
     P(1) = 0                                                            STAT
     P(2) = 1                                                            STAT
     P(3) = -1                                                           STAT
     CALL ACTRST(TM,  NNN, 3, 1, P, 1, Q, SQ)                            STAT
     CALL FFOUT(SQ,S2,N1,N2,FVAL,PRFVAL)                                 STAT
     PRINT 3600,IP2(3), IP2(4), FVAL, PRFVAL                             STAT
     P(1) = 1                                                            STAT
     P(2) = 0                                                            STAT
     CALL ACTRST(TM,  NNN, 3, 1, P, 1, Q, SQ)                            STAT
     CALL FFOUT(SQ,S2,N1,N2,FVAL,PRFVAL)                                 STAT
     PRINT 3600,IP2(5), IP2(6), FVAL, PRFVAL                             STAT
 850 CONTINUE                                                            STAT
 851 NP = 0                                                              STAT
     DO 855 I=1,NT                                                       STAT
     NCOC = NNN(I)                                                       STAT
     DO 855 J=1,NCOC                                                     STAT
     NP = NP + 1                                                         S
 855 ID2(NP) = I                                                         STAT
     IF(ICASE.EQ.2) GO TO 861                                            DEC10
     IF(ICASE.EQ.4) GO TO 860                                            DEC10
```

```
      PRINT 4000                                                        STAT
 4000 FORMAT(1H0/20(4H----)/27H NON-PARAMETRIC STATISTICS.)             STAT
C     CALL  KRUSWAL(X,ID1,ID2,NP,0,0,NT,NNN,LK)                        STAT
      CALL  KRUSWAL(X,ID1,ID2,NP,0,1,NT,NNN,LK)                        STAT
      PRINT 1100,IP2(1)                                                 STAT
 1100 FORMAT(1H0/10HOCONTRAST , A8)                                     STAT
      NP = NNN(1) + NNN(2)                                              STAT
C     CALL  KRUSWAL(X,ID1,ID2,NP,0,0, 2,NNN,LK)                        DEC10
  861 CONTINUE                                                          STAT
      CALL  KRUSWAL(X,ID1,ID2,NP,0,1, 2,NNN,LK)                        STAT
      IF(ICASE.EQ.2) GO TO 999                                          DEC10
      NP = NNN(2) + NNN(3)                                              STAT
      IST = NNN(1) + 1                                                  STAT
      PRINT 1100,IP2(2)                                                 STAT
C     CALL  KRUSWAL(X(IST),ID1(IST),ID2(IST),NP,0,0,2,NNN(2),LK)        STAT
      CALL  KRUSWAL(X(IST),ID1(IST),ID2(IST),NP,0,1,2,NNN(2),LK)        STAT
      NCDC = NNN(3)                                                     STAT
      DO 880 J=1,NCDC                                                   STAT
      K = NNN(1) + J                                                    STAT
      L = NNN(2) + J                                                    STAT
      X(K) = X(L)                                                       STAT
      ID1(K) = ID1(L)                                                   STAT
  880 ID2(K) = ID2(L)                                                   STAT
C     CALL  KRUSWAL(X,ID1,ID2,NP,0,0, 2,NNN,LK)                        DEC10
  860 PRINT 1100,IP2(3)                                                 STAT
      CALL  KRUSWAL(X,ID1,ID2,NP,0,1, 2,NNN,LK)                        DEC10
  999 PRINT 1000,JOB(3)                                                 DEC10
      RETURN                                                            STAT
  200 FORMAT(10X,F12.4,2X,2A1)                                          STAT
  300 FORMAT(13H0GRAND MEAN =,F10.3,  8H     SD =,F10.3,7H     N =,I3)  STAT
  400 FORMAT(I10,F12.4,2X,A4,2X,A2)                                     STAT
  900 FORMAT( 7H0MEAN =,F10.3,  8H     SD =,F10.3,7H     N =,I3//)      STAT
      END                                                               STAT
      SUBROUTINE MESIG(LTEXT,NFIRST,NLAST,SMEAN,SIGMA,NN)               MESIG
      DIMENSION LTEXT(1)                                                MESIG
      NN = 0                                                            MESIG
      SUM = 0.0                                                         MESIG
      SSQ = 0.0                                                         MESIG
      IF(NFIRST - NLAST) 10,90,90                                       MESIG
   10 DO 140 I=NFIRST,NLAST                                             MESIG
      NSLOT = 1                                                         MESIG
      CALL OTEST(I,SUM,SSQ,NN,LTEXT)                                    MESIG
  140 CONTINUE                                                          MESIG
      IF(NN.EQ.0) GO TO 90                                              DEC10
      SMEAN = SUM/NN                                                    MESIG
      SIGMA=0.                                                          DEC10
      IF(NN.EQ.1) RETURN                                                DEC10
      IF((NN*SSQ-SUM*SUM).LE.0.) RETURN                                 DEC10
      SIGMA = ((NN*SSQ - SUM*SUM)/(NN*(NN - 1)))**0.5                   MESIG
      RETURN                                                            MESIG
   90 SMEAN = 0.0                                                       MESIG
      SIGMA = 0.0                                                       MESIG
      RETURN                                                            MESIG
      END                                                               MESIG
      SUBROUTINE OTEST(I,SUM,SSQ,NN,LTEXT)                             OTEST
      DIMENSION LTEXT(1)                                                OTEST
      COMMON XDATA(100),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP        REDUCE
      DATA IZ/4H0.00/, ISS/4HSKIP/, INV/4HINVL/                         OTEST
      DATA IBLANK/4H    /                                               JAN25
      IF(XDATA(I)) 310,410,510                                          OTEST
```

```
  310 IXT = -XDATA(I) + 0.5                                                    OTEST
      IF(IXT - 2) 315,312,315                                                  OTEST
  312 LTEXT(I) = IZ                                                            OTEST
      NN = NN + 1                                                              OTEST
      RETURN                                                                   OTE
  315 LTEXT(I) = ISS                                                           OTE
      RETURN                                                                   OTEST
  410 LTEXT(I) = INV                                                           OTEST
      RETURN                                                                   OTEST
  510 NN = NN + 1                                                              OTEST
      LTEXT(I) = IBLANK                                                        OTEST
      SUM = SUM + XDATA(I)                                                     OTEST
      SSQ = SSQ + XDATA(I)*XDATA(I)                                            OTEST
      RETURN                                                                   OTEST
      END                                                                      OTEST
      SUBROUTINE UERTST                                                        UERTST
      RETURN                                                                   UERTST
      END                                                                      UERTST
      SUBROUTINE  STPLOT(LOOP,NOBST,NSLOT,IWORD,KSTUDY,KTYPE,KMAN,KTEST,STPLOT
     X SMEAN,SIGMA)                                                            STPLOT
      COMMON XDATA(100),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP               REDUCE
      COMMON /HITBLK/LHIT(100),MDATE(100),ISAV(16),MTVOL(100)                  REDUCE
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),          STPLOT
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA                  STPLOT
      DIMENSION YDATE(200),XP(3)                                               REDUCE
      COMMON/RG/ DATESC(3,5),IVV(4), SSMEAN(3), SSIGMA(3), NNN(3)              STPLOT
C     REAL*8 MSTUDY,MTYPE,MMAN,MTEST                                           STPLOT
      REAL   MSTUDY,MTYPE,MMAN,MTEST                                           STPLOT
      REAL KSTUDY, KTYPE, KMAN, KTEST, NITS                                    STPLOT
      DIMENSION NSMAN(7)                                                       STPLOT
      EQUIVALENCE(YDATE,T)                                                     STP
      DATA NSMAN/4,9,8,0,1,2,3/                                                STP
      IF(NOBST - 1) 911,911,1                                                  STPLOT
    1 MFIRST = MDATE(1)                                                        STPLOT
      DO 10 II=1,NOBST                                                         STPLOT
      IF(MDATE(II) - MFIRST) 9,10,10                                           STPLOT
    9 MFIRST = MDATE(II)                                                       STPLOT
   10 YDATE(II) = MDATE(II)                                                    STPLOT
      J1 = NOBST + 1                                                           STPLOT
      J2 = NOBST + 2                                                           STPLOT
      CALL SCALE(XDATA, 3.0,NOBST,1)                                           STPLOT
C     XDATA(NOBST + 1) = 0.0                                                   STPLOT
      IF(XDATA(J2)) 911,911,11                                                 STPLOT
   11 IG = LSTUDY(1)                                                           STPLOT
      YDATE(NOBST + 1) = DATESC(1,IG)                                          STPLOT
      YDATE(NOBST + 2) = 24.0                                                  STPLOT
      UP = (DATESC(2,IG) - DATESC(1,IG))/YDATE(NOBST + 2)                      STPLOT
      DOWN=(DATESC(3,IG) - DATESC(1,IG))/YDATE(NOBST + 2)                      STPLOT
      XP(1) = 0.1                                                              STPLOT
      XP(2) = 0.1+ UP                                                          STPLOT
      XP(3) = 0.1+ DOWN                                                        STPLOT
      NTIC = 3                                                                 STPLOT
      CALL TEXT(LSTUDY(1), LTYPE(1), LMAN(1), LTEST(1),                        STPLOT
     X KSTUDY, KTYPE, KMAN, KTEST, NITS)                                       STPLOT
      KS = KSKIP - KSKIP/3*3                                                   STPLOT
      IF(KS) 208,108,208                                                       STPLOT
C 108 CALL PLOT(4.25,10.5,-3)                                                  STP
  108 CALL PLOT(8.50,10.5,-3)                                                  STP
      FPN = LTYPE(1)                                                           STPLOT
      CALL SYMBOL(0.30,-.4,0.10,KSTUDY    , 0.0, 8)                            STPLOT
```

```
      CALL SYMBOL(1.20,-.4,0.10, KTEST    , 0.0, 8)                          STPLOT
      CALL SYMBOL(2.00,-.4,0.10,4HTYPE, 0.0, 4)                              STPLOT
      CALL NUMBER(2.40,-.4,0.10,FPN,0.0,-1)                                  STPLOT
      CALL SYMBOL(2.60,-.4,0.10,1WORD,0.0,10)                                STPLOT
  208 CALL PLOT(0.0,-3.5,-3)                                                 STPLOT
      CALL AXIS(0.0,0.0,12H JULIAN DATE,                                     STPLOT
     1    -12, 4.0,00.0,YDATE(J1),YDATE(J2),-1,4,3)                          STPLOT
C     CALL AXIS (X,Y,BCD,NC,SIZE,THETA,YMIN,DY,IDEC,NLAB,NTIC)               STPLOT
C     CALL AXIS(0.0,0.0,16H       DATA VALUE,16,3.0,  90.0, Y(J1),Y(J2))     STPLOT
      CALL AXIS(0.0,0.0,NITS    , 8,3.0,  90.0, XDATA(J1),                   STPLOT
     1 XDATA(J2), -1,  4, 1)                                                 STPLOT
      FPN = LMAN(1)                                                          STPLOT
      CALL SYMBOL(0.10,3.01,0.07,KMAN, 0.0, 3)                               STPLOT
      DO 308 JJ=1,3                                                          STPLOT
      KG = 4 - JJ                                                            STPLOT
      CALL NUMBER(XP(KG), 2.89, 0.07, SSMEAN(KG), 0.0, 2)                    STPLOT
  308 CALL NUMBER(XP(KG), 2.77, 0.07, SSIGMA(KG), 0.0, 2)                    STPLOT
      CALL PLOT(UP,0.0,3)                                                    STPLOT
      CALL PLOT(UP,3.0,2)                                                    STPLOT
      CALL PLOT(DOWN,3.0,3)                                                  STPLOT
      CALL PLOT(DOWN,0.0,2)                                                  STPLOT
  700 LL = NSMAN(LMAN(1))                                                    STPLOT
      CALL LINE(YDATE,XDATA,NORST,1, 0,LL)                                   STPLOT
  811 CONTINUE                                                               STPLOT
      KSKIP = KSKIP + 1                                                      STPLOT
  911 RETURN                                                                 STPLOT
      END                                                                    STPLOT
      SUBROUTINE TEXT(LSTUDY,LTYPE,LMAN,LTEST,NSTUDY,NTYPE,NMAN,NTEST,       TEXT
     1 NITS)                                                                 TEXT
C     REAL*8 KSTUDY,MSTUDY(6),KTEST,MURN(124),MELD(24),MITS(124),            TEXT
      REAL   KSTUDY,MSTUDY(5),KTEST,MURN(124),MELD(24),MITS(124),            TEXT
     1MITSE(24),KTYPE,MTYPE(15),MM,KITS                                      TEXT
      REAL NSTUDY, NTYPE, NMAN, NTEST, KMAN, MMAN(9), KITS                   TEXT
      DATA MSTUDY/8HSMEAT   ,8HSKYLAB 2,8HSKYLAB 3,8HSKYLAB 4,8HAPOLLO17/TEXT
     1/                                                                      TEXT
      DATA MMAN/4HCDR ,4HPLT ,4HSPT ,4HC1  ,4HC2  ,4HC3  ,4HC4  ,           TEXT
     X4HC5  ,4HC6  /                                                         TEXT
      DATA (MURN(I),I=1,107)                                                 TEXT
     X           /8HEPI     ,8HNOREPI  ,8HADH     ,8HHYDRO   ,8HALDO    ,TEXT
     X8H17OH     ,8HOSMO    ,8HNA      ,8HK       ,8HMG      ,8HPO4     ,TEXT
     X8HCA       ,8HCL      ,8HH       ,8HSP.GR.  ,8HCREAT   ,8HURICACID,TEXT
     X8HB        ,8HSI      ,8HFE      ,8HAL      ,8HMO      ,8HCU      ,TEXT
     X8HZN       ,8HTI      ,8HNI      ,8HSR      ,8HCR      ,8HBL      ,TEXT
     X8HMN       ,8HLI      ,8HRD      ,8HPD      ,8HAND     ,8HETIO    ,TEXT
     X8HDHEA     ,8H11=0 AND,8H11=0ETIO,8H11OH AND,8H11OHETIO,8HTOTAL   ,TEXT
     X8HLYS      ,8HHIS     ,8HNH3     ,8HARG     ,8HHYP     ,8HASP     ,TEXT
     X8HTHR      ,8HSER     ,8HGLU     ,8HPRO     ,8HGLY     ,8HALA     ,TEXT
     X8HCYS/2    ,8HVAL     ,8HMET     ,8HILE     ,8HLEU     ,8HTYR     ,TEXT
     X8HPHE      ,8HHLYS    ,8HGAMMA-AB,8HORN     ,8HETH     ,8HNH3     ,TEXT
     X8HLYS      ,8H1-CH3HIS,8HHIS     ,8H3-CH3HIS,8HANS     ,8HTRY     ,TEXT
     X8HCRE      ,8HCAR     ,8HARG     ,8HPSER    ,8HPETN    ,8HTYR     ,TEXT
     X8HUREA     ,8HHYP     ,8HASP     ,8HTHR     ,8HSER     ,8HASP NH2 ,TEXT
     X8HGLU NN2  ,8HSAR     ,8HPRO     ,8HGLU     ,8HCIT     ,8HGLC NH2 ,TEXT
     X8HGLY      ,8HALA     ,8HALPHA-AA,8HALPHA-AB,8HVAL     ,8HCYS/2   ,TEXT
     X8HCYT      ,8HMET     ,8HILE     ,8HLEU     ,8HTYR     ,8HPHE     ,TEXT
     X8HB-ALA    ,8HB-AIB   ,8H5HIAA   ,8H        ,8H        ,8H        /JAN25
      DATA (MURN(I),I=108,124)/                                              TEXT
     X8H         ,8H        ,8H        ,8H        ,8H        ,8H        ,TEXT
     X8H         ,8H        ,8HCALORIES,8HPROTEIN ,8HDIET CA ,8HDIET P  ,TEXT
     X8HDIET NA  ,8HDIET MG ,8HDIET K  ,8HDIET H2O,8HWEIGHT  /           TEXT
```

```
      DATA MBLD /8HHYDRO   ,8HALDO    ,8HHGH     ,8HANGIO   ,8HINSULIN ,TEXT
     X8HT4      ,8HACTH    ,8HADH     ,8HPTH     ,8HCAL     ,8HVIT.D.  ,TEXT
     X8HTSH     ,8HOSMO    ,8HNA      ,8HK       ,8HMG      ,8HPO4     ,TEXT
     X8HCA      ,8HCL      ,8HGLU     ,8HTP      ,8HTESTOS  ,8H        ,JAN25
     X8H        /                                                      TEX
      DATA (MITS(I),I=1,107)                                           TEX
     X          /8HUG/TV   ,8HUG/TV   ,8HMU/TV   ,8HUG/TV   ,8HUG/TV  ,TEXT
     X8HMG/TV   ,8HMOSM/TV ,8HMEQ/TV  ,8HMEQ/TV  ,8HMEQ/TV  ,8HMG/TV  ,TEXT
     X8HMEQ/TV  ,8HMEQ/TV  ,8HML0     ,8H        ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML  ,TEXT
     X8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML  ,TEXT
     X8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HTMG/TV ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMMG/TV  ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMMG/TV ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMMG/TV  ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8H        ,8H        ,8H       /JAN25
      DATA (MITS(I),I=108,124)/                                        TEXT
     X8H        ,8H        ,8H        ,8H        ,8H        ,8H       ,TEXT
     X8H        ,8H        ,8HKCAL    ,8HGRAMS   ,8HMG      ,8HMG     ,TEXT
     X8HMG      ,8HMG      ,8HMG      ,8HML      ,8HLBS.    /          TEXT
      DATA MITSB/8HUG/100ML,8HPG/ML   ,8HNG/ML   ,8HMUG/ML/H,8HUU/ML   ,TEXT
     X8HUG/100ML,8H        ,8H        ,8HPG/ML   ,8HPG/ML   ,8HNG/ML   ,TEXT
     X8H        ,8HMOSM/L  ,8HMEQ/L   ,8HMEQ/L   ,8HMG PC   ,8HMG PC  ,JAN
     X8HMG PC   ,8HMG PC   ,8HMG PC   ,8HMG PC   ,8HUG/TV   ,8H        ,JAN
     X8H        /                                                      TEXT
      NSTUDY = MSTUDY(LSTUDY)                                          TEXT
      NMAN = MMAN(LMAN)                                                TEXT
      IF(LTEST - 400) 10,20,20                                         TEXT
   10 KTEST = MBLD(LTEST - 299)                                        TEXT
      KITS = MITSB(LTEST-299)                                          TEXT
      GO TO 30                                                         TEXT
   20 KTEST = MURN(LTEST - 399)                                        TEXT
      KITS = MITS(LTEST-399)                                           TEXT
   30 NTEST = KTEST                                                    TEXT
      NITS = KITS                                                      TEXT
      RETURN                                                           TEXT
      END                                                              TEXT
      SUBROUTINE  KRUSWAL (XX,ID1,ID2,N,NCEN,JOP,K,NU,NC)              KRUSWAL
C     THIS PROGRAM CALCULATES  A  GENERALIZED KRUSKAL-WALLIS K-SAMPLE   KRUSWAL
C     TEST. WHEN K=2, THE TEST IS EQUAL TO GEHAN@S GENERALIZED WILCOXON KRUSWAL
C     TEST. CALCULATIONS FOLLOW THE PROCEDURES GIVEN BY MANTEL (BIOMETRICKRUSWAL
C     MARCH 1967)                                                      KRUSWAL
C                                                                      KRUSWAL
C     IN CASE OF TWO-SAMPLE, IF THE SAMPLE SIZES ARE DIFFERENT, IT IS  KRUSWAL
C     SUGGESTED TO NAME THE ONE WITH LESS OBSERVATIONS SAMPLE 1.       KRUSWAL
C      CALL NKRUSWAL(X,ID1,ID2,NALL,NC,JOP,NCOND,NUNCEN,NCEN)          KRUSWAL
C                                                                      KRUSWAL
C      X(I),I=1,NALL     ARRAY CONTAINING ALL OBS IN THE SAMPLES       KRUSWAL
C       ID1(I)=1         ITH OBS IS UNCENSORED                         KRUSWAL
C             =0         ITH OBS IS CENSORED                           KRUSWAL
C       ID2(I)=J         ITH OBS IS FROM JTH SAMPLE                    KRUSWAL
C       NALL             TOTAL NO OF OBS                               KRUSWAL
```

```
C       NC                TOTAL NO OF CENSORED OBS                          KRUSWAL
C       JOP=1             PRINT INFO                                        KRUSWAL
C          =0             DO NOT PRINT                                      KRUSWAL
C       NCOND             NO OF SAMPLES                                     KRUSWAL
C       NUNCEN(I)         NO OF UNCENSORED OBS IN SAMPLE I                  KRUSWAL
C       NCEN(I)           NO OF CENSORED OBS IN SAMPLE I                    KRUSWAL
        DIMENSION XY(600), ID1(600), ID2(600), NU(6), NC(6)                 KRUSWAL
        DIMENSION R1( 600),R2( 600)                                         KRUSWAL
C                                                                           KRUSWAL
C     MAXIMUM NO. OF SAMPLES = 6.                                           KRUSWAL
C     TOTAL NUMBER OF OBSERVATIONS ALLOWED = 600.                           KRUSWAL
C     K = NO. OF SAMPLES                                                    KRUSWAL
C                                                                           KRUSWAL
C                                                                           KRUSWAL
C     ORDER OBS. IN ASCENDING ORDER                                         KRUSWAL
C                                                                           KRUSWAL
        CALL SOR12(XY,N,ID1,ID2)                                            KRUSWAL
C     COMPUTATION OF R1                                                     KRUSWAL
C        STEPS 1 AND 2 : RANK FROM LEFT TO RIGHT, OMITTING RIGHT CENSORED   KRUSWAL
C                        VALUES. ASSIGN NEXT HIGHER RANK TO RIGHT           KRUSWAL
C                        CENSORED VALUES                                    KRUSWAL
C                                                                           KRUSWAL
        IRANK=0                                                             KRUSWAL
        DO 90I=1,N                                                          KRUSWAL
        IF (ID1(I).EQ.0) GO TO 101                                          KRUSWAL
        IRANK=IRANK+1                                                       KRUSWAL
        R1(I)=IRANK                                                         KRUSWAL
        GO TO 90                                                            KRUSWAL
  101   R1(I)=IRANK+1                                                       KRUSWAL
   90   CONTINUE                                                            KRUSWAL
C                                                                           KRUSWAL
C        STEP 3 : REDUCE  THE RANK OF TIED OBSERVATIONS TO THE LOWEST       KRUSWAL
C                 RANK FOR THE VALUE                                        KRUSWAL
C                                                                           KRUSWAL
        K1=N-1                                                              KRUSWAL
        L1=1                                                                KRUSWAL
   12   IF (XY(L1).NE.XY(L1+1)) GO TO 11                                    KRUSWAL
        JEMP=ID1(L1)*ID1(L1+1)                                              KRUSWAL
        IF(JEMP .EQ. 0) GO TO 11                                            KRUSWAL
        R1(L1+1)=R1(L1)                                                     KRUSWAL
        IF (L1.EQ.K1) GO TO 13                                              KRUSWAL
        L1=L1+1                                                             KRUSWAL
        GO TO 12                                                            KRUSWAL
   11   IF (L1.EQ.K1) GO TO 13                                              KRUSWAL
        L1=L1+1                                                             KRUSWAL
        GO TO 12                                                            KRUSWAL
   13   CONTINUE                                                            KRUSWAL
C                                                                           KRUSWAL
C     COMPUTATION OF R2                                                     KRUSWAL
C        STEP 1 : RANK FROM RIGHT TO LEFT                                   KRUSWAL
C                                                                           KRUSWAL
        DO 14 I=1,N                                                         KRUSWAL
   14   R2(I)=N-I+1                                                         KRUSWAL
C                                                                           KRUSWAL
C        STEP 2 : REDUCE THE RANK OF TIED OBSERVATIONS TO THE LOWEST RANK   KRUSWAL
C                 FOR THE VALUE                                             KRUSWAL
C                                                                           KRUSWAL
        L1=N                                                                KRUSWAL
   22   IF (XY(L1).NE.XY(L1-1)) GO TO 21                                    KRUSWAL
        JEMP=ID1(L1)*ID1(L1-1)                                              KRUSWAL
```

```
      IF (JEMP .EQ. 0) GO TO 21                                          KRUSWAL
      R2(L1-1)=R2(L1)                                                    KRUSWAL
      IF (L1.EQ. 2) GO TO 23                                             KRUSWAL
      L1=L1-1                                                            KRUSWAL
      GO TO 22                                                           KRUSWAL
 21   IF (L1.EQ. 2) GO TO 23                                             KRUSWAL
      L1=L1-1                                                            KRUSWAL
      GO TO 22                                                           KRUSWAL
 23   CONTINUE                                                           KRUSWAL
C                                                                        KRUSWAL
C       STEP 3 : REDUCE THE RANK OF RIGHT CENSORED OBSERVATIONS TO UNITY KRUSWAL
C                                                                        KRUSWAL
      IF (NCEN .EQ. 0) GO To 501                                         KRUSWAL
      DO 24 I=1,N                                                        KRUSWAL
      IF (ID1(I) .EQ. 1) GO TO 24                                        KRUSWAL
      R2(I)=1.                                                           KRUSWAL
 24   CONTINUE                                                           KRUSWAL
C                                                                        KRUSWAL
C     COMPUTE FINAL SCORES -R1(I)                                        KRUSWAL
C                                                                        KRUSWAL
 501  CONTINUE                                                           KRUSWAL
      DO 25 I=1,N                                                        KRUSWAL
 25   R1(I)=R1(I)-R2(I)                                                  KRUSWAL
      IF(JOP.NE.1) GO TO 37                                              KRUSWAL
      PRINT 30                                                           KRUSWAL
 30   FORMAT(1H0,8X,@I@,8X,@OBSERVATIONS@,8X,@SAMPLE@,8X,@SCORES@)       KRUSWAL
      DO 31 I=1,N                                                        KRUSWAL
      IF (ID1(I).EQ.0) GO To 34                                          KRUSWAL
      PRINT 33, I,XY(I),ID2(I),R1(I)                                     KRUSWAL
      GO TO 31                                                           KRUSWAL
   34 PRINT 35, I,XY(I),ID2(I),R1(I)                                     KRUSWAL
 33   FORMAT(1H ,6X,I3,7X,F8.1,15X,I2,12X,F6.0)                          KRUSWAL
 35   FORMAT(1H ,6X,I3,7X,F8.1,1H+,14X,I2,12X,F6.0)                      KRUSWAL
 31   CONTINUE                                                           KRUSWAL
   37 PRINT 36                                                           KRUSWAL
 36   FORMAT(1H0)                                                        KRUSWAL
      IF (K.GT.2) GO TO 200                                              KRUSWAL
      CALL TWOSPL(R1,ID2,N,NU,NC)                                        KRUSWAL
      GO TO 1000                                                         KRUSWAL
 200  CALL AKSPL(K,N,R1,NU,NC,ID2)                                       KRUSWAL
 1000 CONTINUE                                                           KRUSWAL
      RETURN                                                             KRUSWAL
      END                                                                KRUSWAL
      SUBROUTINE SORT2(X,N,ID,IC)                                        SORT2
      DIMENSION X(1),ID(1),IC(1)                                         SORT2
      DO 1 I=1,N                                                         SORT2
      J=N-I+1                                                            SORT2
      JJ=J-1                                                             SORT2
      IF (JJ .LT. 1)GO TO 1                                              SORT2
      DO 2 K=1,JJ                                                        SORT2
      IF (X(K).NE.X(J)) GO TO 3                                          SORT2
      IF (ID(K)-ID(J)) 4,2,2                                             SORT2
 3    IF (X(K) .LT. X(J)) GO TO 2                                        SORT2
 4    X1=X(J)                                                            SORT2
      ITEM=ID(J)                                                         SORT2
      ICTE=IC(J)                                                         SORT2
      X(J)=X(K)                                                          SORT2
      ID(J)=ID(K)                                                        SORT2
      IC(J)=IC(K)                                                        SORT2
      X(K)=X1                                                            SORT2
```

```
      ID(K)=ITEM                                                        SORT2
      IC(K)=ICTE                                                        SORT2
    2 CONTINUE                                                          SORT2
    1 CONTINUE                                                          SORT2
      RETURN                                                            SORT2
      END                                                               SORT2
      SUBROUTINE TWOSPL(R1,ID2,N,NU,NC)                                 TWOSPL
      DIMENSION ID2(1),R1(1),NU(1),NC(1)                                TWOSPL
C     DATA ONE,TWO/@ONE@,@TWO@/                                         TWOSPL
      DATA ONE/10HONE       /                                           TWOSPL
      DATA TWO/10HTWO       /                                           TWOSPL
      WW=0.                                                             TWOSPL
      DO 26 I=1,N                                                       TWOSPL
      IF (ID2(I).EQ.2) GO TO 26                                         TWOSPL
      WW=WW+R1(I)                                                       TWOSPL
   26 CONTINUE                                                          TWOSPL
      SUM=0.                                                            TWOSPL
      DO 27 I=1,N                                                       TWOSPL
   27 SUM=SUM+R1(I)**2                                                  TWOSPL
      XN1=NU(1)+NC(1)                                                   TWOSPL
      XN2=NU(2)+NC(2)                                                   TWOSPL
      XN=N                                                              TWOSPL
      VAR=XN1*XN2*SUM/(XN*(XN-1.))                                      TWOSPL
      VAR=SQRT(VAR)                                                     TWOSPL
C                                                                       TWOSPL
C     CALCULATE CONTINUITY CORRECTION                                   TWOSPL
C                                                                       TWOSPL
      SIGN=-1.                                                          TWOSPL
      IF (WW.LT.0.) SIGN=1.                                             TWOSPL
      COR=.5                                                            TWOSPL
      IF (FLOAT((NC(1)+NC(2))/N) .LT. 0.2) COR=1.0                      TWOSPL
      COR=SIGN*COR                                                      TWOSPL
      IF (WW.EQ.0.) COR=0.                                              TWOSPL
C                                                                       TWOSPL
C     CALCULATE FINAL W SCORE AND PROBABILITY.                          TWOSPL
C                                                                       TWOSPL
      WSCORE=(WW+COR)/VAR                                               TWOSPL
      WSC=WSCORE                                                        TWOSPL
      IF (ABS(WSCORE)-3.1) 305,305,302                                  TWOSPL
  305 P1=100.*(1.-PROB(WSC))                                            TWOSPL
      GO TO 301                                                         TWOSPL
  302 P1=0.1                                                            TWOSPL
  301 CONTINUE                                                          TWOSPL
      P2=P1*2.                                                          TWOSPL
      P2=AMIN1(P2,100.)                                                 TWOSPL
      PRINT 600, WW,VAR,WSCORE                                          TWOSPL
  600 FORMAT(1H0,3X,1HW,7X,8HST. DEV.,3X,10HASYMPTOTIC/25X,6HWSCORE/1X,FTWOSPL
     19.0,F8.2,7X,F5.2)                                                 DEC10
      PRINT 601, WSCORE,P2,TWO                                          TWOSPL
      PRINT 601, WSCORE,P1,ONE                                          TWOSPL
  601 FORMAT(1X,F5.2,1X,@IS SIGNIFICANT AT THE @, F7.1,@ PERCENT LEVEL -TWOSPL
     1@,A3,@ TAILED TEST@)                                              TWOSPL
      PRINT 602                                                         TWOSPL
  602 FORMAT(1H0)                                                       TWOSPL
      RETURN                                                            TWOSPL
      END                                                               TWOSPL
      SUBROUTINE AKSPL(K,N,R1,NU,NC,ID2)                                AKSPL
      DIMENSION R1(1),NU(1),NC(1),W(6),ID2(1)                           AKSPL
C                                                                       AKSPL
C     TEST STATISTIC FOR K(GREATER THAN 2)-SAMPLE CASE                  AKSPL
```

```
C                                                                       AKSPL
      DO 201 I=1,K                                                      AKSPL
 201  W(I)=0.                                                           AKSPL
      DO 202 IJ=1,N                                                     AKSPL
      I=ID2(IJ)                                                         AKSPL
 202  W(I)=W(I)+R1(IJ)                                                  AKSPL
      T=0.                                                              AKSPL
      DO 203 IJ=1,N                                                     AKSPL
 203  T=T+R1(IJ)**2                                                     AKSPL
      B=0.                                                              AKSPL
      PRINT 300                                                         AKSPL
 300  FORMAT(1H0,@SAMPLE@,8X,@W(I)@,10X,@N(I)@)                         AKSPL
      DO 204 I=1,K                                                      AKSPL
      UC=NU(I)+NC(I)                                                    AKSPL
      PRINT 301, I,W(I),UC                                              AKSPL
 301  FORMAT(1H ,3X,I1,7X,F8.0,7X,F5.0)                                 AKSPL
 204  B=B+(W(I)**2/UC)                                                  AKSPL
      PRINT 302, B,T                                                    AKSPL
 302  FORMAT(1H0,@B =@,F12.2,5X,@T =@,F10.0)                            AKSPL
      WSCORE=(B/T)*FLOAT(N-1)                                           AKSPL
C                                                                       AKSPL
C     WSCORE HAS CHI-SQUARE DISTRIBUTION WITH (K-1) D. F.               AKSPL
C                                                                       AKSPL
      WS=WSCORE                                                         AKSPL
      XK=K-1                                                            AKSPL
      XM=WS/2.                                                          AKSPL
      IC=XK/2.                                                          AKSPL
      SWS=SQRT(WS)                                                      AKSPL
      EM=1./EXP(XM)                                                     AKSPL
      PRINT 205, WSCORE                                                 AKSPL
 205  FORMAT(1H0,@WSCORE = @,F7.3)                                      AKSPL
      XK2=XK/2.                                                         AKSPL
      IF ((XK2-FLOAT(IC)).NE.0) GO TO 500                               AKSPL
      SUM=0.                                                            AKSPL
      PROD=1.                                                           AKSPL
      DO 1 I=1,IC                                                       AKSPL
      IF (I.GT.1) GO TO 2                                               AKSPL
      XI=1                                                              AKSPL
      GO TO 3                                                           AKSPL
 2    XI=I-1                                                            AKSPL
 3    PROD=XI*PROD                                                      AKSPL
      TERM=XM**(I-1)/PROD                                               AKSPL
 1    SUM=SUM+TERM                                                      AKSPL
      CPROB=SUM*EM                                                      AKSPL
      P=100.*CPROB                                                      AKSPL
      IF (P.LT.0.1000) GO TO 10                                         AKSPL
      PRINT 206, WSCORE,P                                               AKSPL
 206  FORMAT(1H0,F7.3,@ IS SIGNIFICANT AT THE @,F5.1,@ PERCENT LEVEL@)  AKSPL
      GO TO 1000                                                        AKSPL
 10   P=0.10                                                            AKSPL
      PRINT 12, WSCORE,P                                                AKSPL
 12   FORMAT(1H0,F7.3,@ IS SIGNIFICANT WITH PROBABILITY LESS THAN @,F7.3AKSPL
     1,@ PERCENT LEVEL@)                                                AKSPL
      GO TO 1000                                                        AKSPL
 500  GAMH=1.7724538509                                                 AKSPL
      TERM=SQRT(XM)/(0.5*GAMH)                                          AKSPL
      SUM=TERM                                                          AKSPL
      NR=(XK-3.)/2.                                                     AKSPL
      IF(NR.EQ.0) GO TO 502                                             AKSPL
      DO 501 I=1,NR                                                     AKSPL
```

```
      X1=I                                                              AKSPL
      TERM=TERM*2.*XM/(2.*XI+1.)                                        AKSPL
  501 SUM=SUM+TERM                                                      AKSPL
  502 CHISQ=(SUM*EM)*100.                                               AKSPL
      P2=100.*2.*(1.-PROB(SWS))                                         AKSPL
      P=P2+CHISQ                                                        AKSPL
      IF (P.LT.0.0005) GO TO 14                                         AKSPL
      PRINT 206,  WSCORE,P                                              AKSPL
      GO TO 1000                                                        AKSPL
   14 P=0.001                                                           AKSPL
      PRINT 12,  WSCORE,P                                               AKSPL
 1000 CONTINUE                                                          AKSPL
      RETURN                                                            AKSPL
      END                                                               AKSPL
      FUNCTION PROB(X)                                                  PROB
C                                                                       PROB
C       THIS FUNCTION ROUTINE COMPUTES                                  PROB
C                    DISTRIBUTION FUNCTION(X)        IF X GE 0          PROB
C                1 - DISTRIBUTION FUNCTION(X)        IF X LT 0          PROB
C       OF A S1. NORMAL VARIABLE USING APPROXIMATION 26.2.19  P.932     PROB
C       HANDBOOK OF MATH. FUNCTIONS                                     PROB
      DATA D1,D2,D3,D4,D5,D6/.0498673470,.0211410061,.0032776263,.00003PROB
     10036,.0000488906,.0000053830/                                     PROB
      IF (X) 20,30,40                                                   PROB
   30 PROB=0.5                                                          PROB
      RETURN                                                            PROB
   20 X1=-X                                                             PROB
      GO TO 50                                                          PROB
   40 X1=X                                                              PROB
   50 A=1.+X1*(D1+X1*(D2+X1*(D3+X1*(D4+X1*(D5+X1*D6)))))                PROB
      PROB=1.-0.5*A**(-16)                                              PROB
   60 CONTINUE                                                          PROB
      RETURN                                                            PROB
      END                                                               PROB
      SUBROUTINE FFOUT(S1, S2, N1, N2,FDAT, PRB)                        FFOUT
      FDAT = (S1/N1)/(S2/N2)                                            FFOUT
      PRB= FISH(FDAT,N1,N2)                                             FFOUT
      PRB = (1.0 - PRB)*100.                                            FFOUT
      IF(PRB - 0.1) 1,2,2                                               FFOUT
    1 PRB = 0.1                                                         FFOUT
    2 RETURN                                                            FFOUT
      END                                                               FFOUT
      FUNCTION FISH(F,N1,N2)                                            FISH
      LOGICAL E1,E2,E3                                                  FISH
       IF(N1.GE.100.AND.N2.GE.100) GOTO 9                               FISH
C-----------------------------------------------------------------------FISH
C     INITIALIZATION AND SETTING OF LOGICAL SWITCHES  TO .TRUE. IF      FISH
C     THE DEGREES OF FREEDOM ARE EVEN .                                 FISH
C-----------------------------------------------------------------------FISH
      E1=.FALSE.                                                        FISH
      E2=.FALSE.                                                        FISH
      E3=.FALSE.                                                        FISH
      IF(MOD(N1,2).EQ.0) E1=.TRUE.                                      FISH
      IF(MOD(N2,2).EQ.0) E2=.TRUE.                                      FISH
      X=N2/(N2+N1*F)                                                    FISH
      IF(.NOT.(E1.OR.E2)) GO TO 5                                       FISH
      IF(E1.AND..NOT.E2) GO TO 1                                        FISH
      IF(.NOT.E1.AND.E2) GO TO 2                                        FISH
      IF(N1.LE.N2) GO TO 1                                              FISH
C-----------------------------------------------------------------------FISH
```

```
C   INITIALIZATION FOR SECOND DEGREE OF FREEDOM EVEN  AND LESS THAN     FISH
C      FIRST DEGREE OF FREEDOM IF IT TOO IS EVEN                        FISH
C-----------------------------------------------------------------------FISH
    2 I=N1                                                              FISH
      N1=N2                                                             FISH
      N2=I                                                              FISH
      X=1.0-X                                                           FISH
      E3=.TRUE.                                                         FISH
C-----------------------------------------------------------------------FISH
C      INITIALIZATION FOR FIRST DEGREE OF FREEDOM EVEN AND LESS THAN    FISH
C   SECOND DEGREE OF FREEDOM IF IT IS EVEN                              FISH
C-----------------------------------------------------------------------FISH
    1 Y=1.0-X                                                           FISH
C-----------------------------------------------------------------------FISH
C      CALCULATION OF PROBABILITY FOR AT LEAST ONE DEGREE OF FREEDOM EVENFISH
C-----------------------------------------------------------------------FISH
      FISH=0.0                                                          FISH
      H=SQRT(X**N2)                                                     FISH
      M=N1/2-1                                                          FISH
      MCDC = M + 1                                                      FISH
      DO 3 ICDC=1,MCDC                                                  FISH
      I = ICDC - 1                                                      FISH
      FISH=FISH+H                                                       FISH
    3 H=(H*Y*(N2+2.*I))/(2.*(I+1.))                                     FISH
      IF(E3) GO TO 4                                                    FISH
C-----------------------------------------------------------------------FISH
C        ADJUST CALCULATED PROBABILITY IF ITS ONES COMPLEMENT WAS       FISH
C    CALCULATED ORIGINALLY                                              FISH
C-----------------------------------------------------------------------FISH
      FISH=1.0-FISH                                                     FISH
      RETURN                                                            FISH
    4 I=N1                                                              FISH
      N1=N2                                                             FISH
      N2=I                                                              FISH
      RETURN                                                            FISH
C-----------------------------------------------------------------------FISH
C      CALCULATION OF THE PROBABILITY FOR BOTH DEGREES OF FREEDOM ODD   FISH
C-----------------------------------------------------------------------FISH
    5 Y=1.0-X                                                           FISH
      H=.63661977*SQRT(X*Y)                                             FISH
      FISH=.63661977*ACOS(SQRT(X))                                      FISH
      IF(N2.EQ.1) GO TO 8                                               FISH
      M=N2-2                                                            FISH
      DO  6  I=1,M,2                                                    FISH
      FISH=FISH+H                                                       FISH
    6 H=H*X*(I+1)/(I+2)                                                 FISH
    8 IF(N1.EQ.1) RETURN                                                FISH
      H=H*N2                                                            FISH
      M=N1-2                                                            FISH
      DO  7  I=1,M,2                                                    FISH
      FISH=FISH-H                                                       FISH
    7 H=H*Y*(N2+I)/(I+2)                                                FISH
      RETURN                                                            FISH
    9  D1=N1                                                            FISH
       D2=N2                                                            FISH
       DT=(D1/D2)*F                                                     FISH
       DN=SQRT((2.*D2-1.)*DT)-SQRT(2.*D1-1.)                            FISH
       X=DN/SQRT(1.+DT)                                                 FISH
       FISH=PHI(X)                                                      FISH
       RETURN                                                           FISH
```

```
      END                                                                FISH
      REAL FUNCTION PHI(X)                                               PHI
C-----------------------------------------------------------------------PHI
C                                                                        PHI
C  PHI CALCULATES THE AREA UNDER THE NORMAL CURVE                        PHI
C  A TRANSFORMATION AND J-FRACTION ARE USED ( SEE METHOD )               PHI
C                                                                        PHI
C-----------------------------------------------------------------------PHI
      LOGICAL UPPER                                                      PHI
      IF(X.LT.-13.27) GO TO 6                                            PHI
      IF(X.GT.8.5) GO TO 8                                               PHI
      IF(X.NE.0.0) GO TO 2                                               PHI
      PHI=0.50                                                           PHI
      RETURN                                                             PHI
    2 UPPER=X.GT.0.0                                                     PHI
      Z=    (ABS(X))                                                     PHI
      Y= 5.6418953027302E-1* EXP(- Z*Z /2.E0)                            PHI
      Z = Z /1.4142135623731E0                                           PHI
      T=0.0E0                                                            PHI
      IF( ABS(Y/Z).GT.0.0) T=Y/(Z-6.9183675618730E-6                     PHI
     1+5.0025350900390E-1/(Z+1.2386797611409E-2+7.7267300865878E-1       PHI
     2/(Z-4.3263982143053E0 +7.3456287718055E1/(Z+1.5040871364290E1      PHI
     3+6.2066245657235 6E0 /(Z+8.8971612130791E0 +4.9182171845874E1      PHI
     4/(Z-2.5108230069509E0 -2.8225972942737E0/(Z-9.7597917308472E-1     PHI
     5+2.4244213526837E1 /(Z+4.8008570125081E0 +4.9227853919002E-1       PHI
     6/(Z+7.6621170927661E0 +5.0285619125788E1/(Z-4.6529284984655E0      PHI
     7  ))))))))))                                                     PHI
      T=T/2.E0                                                           PHI
      IF(UPPER) GO TO 4                                                  PHI
      PHI=T                                                              PHI
      RETURN                                                             PHI
    4 PHI=1.0E0-T                                                        PHI
      RETURN                                                             PHI
    6 PHI = 0.0                                                          PHI
      RETURN                                                             PHI
    8 PHI = 1.0                                                          PHI
      RETURN                                                             PHI
      END                                                                PHI
      SUBROUTINE ERROR                                                   ERROR
      COMMON /SISBUF/ IBUF(35),IUR(125),IBL(35),IBUFF(30),KEE,KEF,       ERROR
     +                MFUNC,IFILE,IERR                                   ERROR
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)            ERROR
      GO TO(100,200,300,400) IFILE                                       ERROR
  100 IERR = IFETCH(FITRA1,3LIRS)                                        ERROR
      CALL STOREF(FITRA1,3LIRS,0)                                        ERROR
      GO TO 500                                                          ERROR
  200 IERR = IFETCH(FITRA2,3LIRS)                                        ERROR
      CALL STOREF(FITRA2,3LIRS,0)                                        ERROR
      GO TO 500                                                          ERROR
  300 IERR = IFETCH(FITRA3,3LIRS)                                        ERROR
      CALL STOREF(FITRA3,3LIRS,0)                                        ERROR
      GO TO 500                                                          ERROR
  400 IERR = IFETCH(FITRA4,3LIRS)                                        ERROR
      CALL STOREF(FITRA4,3LIRS,0)                                        ERROR
  500 IF(IERR.EQ.0) RETURN                                               ERROR
      PRINT 601,IFILE,IERR,KEE,KEF                                       ERROR
      RETURN                                                             ERROR
  601 FORMAT(* RA*,I1,* ERROR *,O10,2I10)                                ERROR
      END                                                                ERROR
```

D.4 Control cards and modification deck for modifying program RETD to perform analysis of variance

```
NASARET,CM20000,T1000.                    NEWTON
ACCOUNT,AN12318,,.
GET,RTREVE.
GET(ANFILE)
RFL,70000.
MODIFY,F,P=RTREVE,LO=CET.
SETCORE(0)
FTN,I,L=0,PL=50000,T.
ATTACH,FTNIMSL/UN=LB12345.
GET,FTNPLT/UN=AN12005.
GET,RA1.
GET,RA2.
GET,RA3.
GET,RA4.
RFL,110000.
MODE(1)
LOAD,LGO,FTNIMSL,FTNPLT.
EXECUTE.
PACK,ZZZZZEF.
COPYSBF,ZZZZZEF,OUTPUT.
EXIT.
PACK,ZZZZZEF.
COPYSBF,ZZZZZEF,OUTPUT.
-
*CREATE  ANFILL
*IDENT  APR24
*DECK  STAT
*D  75
      IVV(1)=IVV(2)=IVV(3)=0
*D  108
      IF(IFIRST-ILAST)162,163,166
  163 PRINT 401
  401 FORMAT(*  SINGLE VALUE NOT ANALYZED*)
*DECK  MESIG
*D  6
      IF(NFIRST-NLAST) 10,10,90
*IDENT  SETWRK
*DECK  KRUSWAL
*D  1
      SUBROUTINE KRUSWAL(XIN,ID1,ID2IN,N,NCEN,JOP,K,NU,NC,IP2)
*D  22
      DIMENSION XIN(1),ID1(1),ID2IN(1),NU(6),NC(6),XY(120),ID2(120)
*I  28
C     MOVE INCOMMING ARRAYS TO WORK ARRAYS
      DO 10 I=1,N
      XY(I)=XIN(I)
      ID2(I)=ID2IN(I)
   10 CONTINUE
*IDENT  FIX
*DECK RETD
*D  1
      PROGRAM RETD(INPUT,OUTPUT,TAPE6=OUTPUT)
*DECK  STAT
*MODNAME  DEC10
*D  8,19
*D  23
*D  25
*D  26
      IF(ICASE.EQ.2) GO TO 998
*MODNAME STAT
```

```
*I 190
      IF(ICASE.EQ.4) 710,711
  710 IF(NNN(2).EQ.1) GO TO 712
      GO TO 713
  712 NNN1=NNN(1)+1
      NSLTM1=J-1
      DO 179 I=NNN1,NSLTM1
  179 X(I)=X(I+1)
  713 NNN(2)=NNN(3)
      NNN(3)=0
  711 CONTINUE
*I  219
      P(3)=0
*D  222,223
      PRINT 3600,IP2(1),FVAL,PRFVAL
 3600 FORMAT(1H0,A8,F8.3,23H IS SIGNIFICANT AT THE    ,F5.1,
*D  230
      PRINT 3600,IP2(2),FVAL,PRFVAL
*D  235
      PRINT 3600,IP2(3),FVAL,PRFVAL
*D  262
      L=NNN(1)+NNN(2)+J
*I 265
      NP=NNN(1)+NNN(3)
      NNN(2)=NNN(3)
*I  267
  998 CONTINUE
*IDENT FIX2
*DECK  STAT
*I  155
      NN=NN-NNN(2)
*I  165
      NN=NN-NNN(3)
*D  247
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,NT,NNN,LK,LK)
*D  252
      IPI=2
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,2,NNN,LK,IPI)
*D  257
      IPI=3
      CALL KRUSWAL(X(IST),ID1(IST),ID2(IST),NP,0,1,2,NNN(2),LK,IPI)
*D  267
      IPI=3
      IF(ICASE.EQ.4) IPI=2
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,2,NNN,LK,IPI)
*DECK  KRUSWAL
*D 113
      CALL TWOSPL(R1,ID2,N,NU,NC,IP2)
*DECK  TWOSPL
*D  1
      SUBROUTINE TWOSPL(R1,ID2,N,NU,NC,IP2)
*D  8
      IF(ID2(I).EQ.IP2) GO TO 26
*IDENT  XREDUC
*DECK RETD
*MODNAME  REDUCE
*D  1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D  2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
```

```
*DECK  RET
*MODNAME  REDUCE
*D  1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D  2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*D  3
      IF(IH.LE.151)  GO TO 10
*D  4
      IF(NSMP.LE.150)  GO TO 35
*DECK  SETCRI
*MODNAME  REDUCE
*D  1
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*DECK  RETRVE
*MODNAME  REDUCE
*D  1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D  2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*D  3
      IF(M.GT.150)  GO TO 110
*DECK  STAT
*MODNAME  REDUCE
*D  1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D  2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*DECK  OTEST
*MODNAME  REDUCE
*D  1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*DECK  STPLOT
*MODNAME  REDUCE
*D  1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D  2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*IDENT ANOV1
*DECK  STAT
*I  6
      COMMON Y(3,3,80),N(3,3),A(4,4),B(4,4),AP(3),Q(3),W(3),V(3),WW(3),
     1VV(3),NSUMI(3),SUMYI(3),NSUMJ(3),SUMYJ(3),SUMYIJ(3,3),TAUHAT(3),
     2    BETHAT(3),THAT(3),BHAT(3)
      COMMON/AOV2NE/IDEN(8),NAMET,NAMEB,IWT,SUMSQX,TAU,MNSQT,FRATIO,
     1 IERR,ERROR,MNSQE,SST,MNSST,FSST,INT,TIN,TINT,FINT,ITHT,WITHT,
     2 WIT,NSUM,SUMSQY,SUMSQT,BETA,MNSQB,FRATB,ERRORB,MNSQBE,SSB,MNSSB,
     3 FSSB,BIN,BINT,FINB,PFRAT,PFSST,PFINT,PFRATB,PFSSB,PFINB
*I  7
      REAL MNSSB,MNSST,MNSQR,MNSQBE,MNSQT,MNSQE
      DIMENSION DEN(8)
      DIMENSION KOE(3)
      EXTERNAL TWOWT,TWOAVT,TWOAVB
*I  12
      EQUIVALENCE (IDEN,DEN)
*MODNAME  FIX
*I  10
      ITOT=JTOT=3
      IF(ICASE.EQ.2.OR.ICASE.EQ.4) ITOT=2
      IB=LMAN(1)
```

```
      K1=0
      DO 699 IT=1,ITOT
      K=N(IT,IB)=NNN(IT)
      DO 702 I=1,K
  702 Y(IT,IB,I)=X(I+K1)
      K1=K+K1
  699 CONTINUE
*MODNAME  FIX
*I  20
      IF(IB.EQ.3) 701,999
  701 MAXT=MAXB=3
      MAXT1=MAXT+1
      MAXB1=MAXB+1
      NAMET=4HTIME
      NAMEB=5HASTRO
      IWT=1
      DEN(1)=KSTUDY
      DEN(2)=KTEST
      IDEN(8)=3HALL
      IF(ICASE.EQ.2) IDEN(8)=IP2(1)
      IF(ICASE.EQ.4) IDEN(8)=IP2(3)
      DO 703 J=1,JTOT
      DO 704 I=1,ITOT
      KK=N(I,J)
      PRINT 705,(Y(I,J,K),K=1,KK)
  705 FORMAT(F12.4)
  704 CONTINUE
  703 CONTINUE
      PRINT 1000,JOB(3)
      CALL ANOV2NE(MAXT,MAXB,ITOT,JTOT,A,B,AP,Q,N,W,V,WW,VV,Y,NSUMI,
     1  SUMYI,NSUMJ,SUMYJ,SUMYIJ,TAUHAT,BETHAT,THAT,BHAT,SUMIJ,MAXT1,
     2   MAXB1)
      IF(IWT.LT.0) GO TO 999
      ALPHA=.05
  797 FORMAT(/////@0 RANKED MEAN CONTRASTS(S-METHOD) ON @,A8,
     2       @ ASSUMING NO INTERACTION@/
     3       @ AT THE @,F5.3,@ SIGNIFICANCE LEVEL @///)
  798 FORMAT(@1@,8A10)
  799 FORMAT(/////@0 RANKED MEAN CONTRASTS(S-METHOD) ON @,A8,
     2       @ AT THE @,F5.3,@ SIGNIFICANCE LEVEL @///)
      WRITE(6,798) IDEN
      WRITE(6,797) NAMET,ALPHA
      CALL SCHEFE(ITOT,TAUHAT,WIT,ITHT,ALPHA,KOE,WW,TWOAVT)
      WRITE(6,799) NAMET,ALPHA
      CALL SCHEFE(ITOT,THAT,WIT,ITHT,ALPHA,KOE,WW,TWOWT)
      WRITE(6,797) NAMEB,ALPHA
      CALL SCHEFE(JTOT,BETHAT,WIT,ITHT,ALPHA,KOE,VV,TWOAVB)
      WRITE(6,799) NAMEB,ALPHA
      CALL SCHEFE(JTOT,BHAT,WIT,ITHT,ALPHA,KOE,VV,TWOWT)
-
JOB  DATE   300
STUDYEACH     3
DATE ALL
TYPE EACH     1
TEST EACH   436
END
EOF
-
JOB  DATE  3300
TEST RANGE  410  412
```

D.5 Analysis of Variance program as contained in file ANFILE

```
      SUBROUTINE ANOV2NE (MAXT,MAXB,ITOT,JTOT,A,B,P,Q,N,W,V,WW,VV,Y,     ANO
     2                    NSUMI, SUMYI,NSUMJ,SUMYJ,SUMYIJ,TAUHAT,BETHAT,THAT,  ANO
     3                    BHAT,SUMTOT,MAXT1,MAXB1)                          ANO
C                                                                          ANO
C          TWO-WAY ANALYSIS OF VARIANCE WITH UNEQUAL NUMBERS PER CELL      ANO
C                                                                          ANO
C          MODEL                                                           ANO
C                                                                          ANO
C     Y(I,J,K) = MU + TAU(I) + BETA(J) + INTERACTION(I,J) + ERROR(I,J,K)   ANO
C                                                                          ANO
C                                                                          ANO
C          MODEL ASSUMING ADDITIVITY                                       ANO
C                                                                          ANO
C     Y(I,J,K) = MU + TAU(I) + BETA(J) + ERROR(I,J,K)                      ANO
C                                                                          ANO
C                                                                          ANO
C          SUBROUTINE ARGUMENTS                                            ANO
C                                                                          ANO
C IN   MAXT          MAXIMUM FIRST DIMENSION OF Y,A,N,SUMYIJ               ANO
C IN   MAXB          MAXIMUM SECOND DIMENSION OF Y,FIRST OF B              ANO
C IN   ITOT          NUMBER OF TAU EFFECTS                                 ANO
C IN   JTOT          NUMBER OF BETA EFFECTS                                ANO
C      A             MAXT BY MAXT WORK ARRAY --- TAU EFFECTS  A*TAUHAT = G ANO
C      B             MAXB BY MAXB WORK ARRAY ---BETA EFFECTS  B*BETHAT = P ANO
C      P             MAXT WORK VECTOR                                      ANO
C      Q             MAXB WORK VECTOR                                      ANO
C IN   N             MAXT BY MAXB (ACTUAL ITOT BY JTOT) NUMBERS OF SAMPLES ANO
C                              PER CELL                                    ANO
C      W             ITOT WEIGHTS OF TAU EFFECTS  (SEE IWT IN COMMON)      ANO
C      V             JTOT WEIGHTS OF BETA EFFECTS (SEE IWT IN COMMON)      ANO
C      VV            MAX(ITOT,JTOT) WORK VECTOR FOR WEIGHTS                ANO
C IN   Y             ITOT BY JTOT BY N(I,J) ARRAY OF DATA                  ANO
C      NSUMI         JTOT VECTOR OF SUMS OF N(I,J) -- SUMED ON I           ANO
C      SUMYI         JTOT VECTOR OF SUMS OF Y(I,J,K) -- SUMED ON K AND I   ANO
C      NSUMJ         ITOT VECTOR OF SUMS OF N(I,J) -- SUMED ON J           ANO
C      SUMYJ         ITOT VECTOR OF SUMS OF Y(I,J,K) -- SUMED ON K AND J   ANO
C      SUMYIJ        ITOT BY JTOT ARRAY OF SUM OF Y(I,J,K) -- SUMED ON K   ANO
C      TAUHAT        ITOT VECTOR OF ESTIMATES OF TAU EFFECTS               ANO
C      BETHAT        JTOT VECTOR OF ESTIMATES OF BETA EFFECTS              ANO
C      X1            MAX(ITOT,JTOT) WORK VECTOR                            ANO
C          COMMON BLOCK AOV2NE ARGUMENTS                                   ANO
C IN   IDEN          10 LENGTH VECTOR CONTAINING AOV TABLE TITLE (40 ALPHA) ANO
C IN   NAMET         2 LENGTH VECTOR CONTAINING THE NAME OF THE TAU EFFECTS ANO
C IN   NAMEB         2 LENGTH VECTOR CONTAINING THE NAME OF THE BETA EFFECTSANO
C IN   IWT           IS 1, IF WEIGHTS W,V ARE TO BE 1/ITOT AND 1/JTOT, RESP.ANO
C                       2, IF W,V ARE TO BE 1/NSUMJ(I),1/NSUMI(J),RESP.    ANO
C                       3, IF W,V CONTAIN ON CALL TO ANOVA2NE, THE WEIGHTS ANO
C                          TO BE USED                                      ANO
C                                                                          ANO
C     THE REMAINING COMMON BLOCK VARIABLES ARE VARIABLES CALCULATED IN     ANO
C         ONE OF THE TWO ANALYSIS OF VARIANCE TABLES PREPARED BY ANOVA2NE  ANO
C                                                                          ANO
C     AOV TABLE ONE CALCULATES WITH THE TAU EFFECTS ADJUSTED FOR UNEQUAL   ANO
C         CELL NUMBERS.  MNSQT,MNSQE,MNSST ARE TYPED REAL AND NOT INTEGE   ANO
C                                                                          ANO
C       S.V.                    D.F.        S.S.        M.S.        F-RATIO ANO
C                                                                          ANO
C       MU,BETA                 JTOT        SUMSQX                          ANO
C                                                                          ANO
C       TAU (ADJ.)            ITOT-1        TAU         MNSQT         FRATIO ANO
```

```
C       NO INTERACTION                                                  ANOV
C                                                                       ANOV
C       ERROR                  IERR       ERROR      MNSGE            ANOV
C       NO INTERACTION                                                  ANOV
C                                                                       ANOV
C       TAU (ADJ.)           ITOT-1        SST        MNSST     FSST   ANOV
C                                                                       ANOV
C       INTERACTION            INT         TIN         TINT     FINT   ANOV
C       EFFECT                                                          ANOV
C                                                                       ANOV
C       WITHIN CELL            ITHT        WITHT       WIT             ANOV
C       ERROR                                                           ANOV
C                                                                       ANOV
C       TOTAL                  NSUM        SUMSQY                      ANOV
C                                                                       ANOV
C                                                                       ANOV
C     AOV TABLE TWO CALCULATES WITH THE BETA EFFECTS ADJUSTED FOR       ANOV
C        UNEQUAL CELL NUMBERS.  MNSQB,MNSQBE,MNSSB ARE TYPED REAL       ANOV
C                                                                       ANOV
C       S.V.                   D.F.        S.S.        M.S.     F-RATIO ANOV
C                                                                       ANOV
C       MU,TAU                 ITOT        SUMSQT                      ANOV
C                                                                       ANOV
C       BETA (ADJ.)          JTOT-1        BETA       MNSQB     FRATB  ANOV
C       NO INTERACTION                                                  ANOV
C                                                                       ANOV
C       ERROR                  IERR        ERRORB     MNSQBE           ANOV
C       NO INTERACTION                                                  ANOV
C                                                                       ANOV
C       BETA (ADJ.)          JTOT-1        SSB        MNSSB     FSSB   ANOV
C                                                                       ANOV
C       INTERACTION            INT         BIN         BINT     FINB   ANOV
C       EFFECT                                                          ANOV
C                                                                       ANOV
C       WITHIN CELL            ITHT        WITHT       WIT             ANOV
C       ERROR                                                           ANOV
C       TOTAL                  NSUM        SUMSQY                      ANOV
C                                                                       ANOV
C                                                                       ANOV
      DIMENSION Y(MAXT,MAXB,1),N(MAXT,1),NSUM1(1),NSUMJ(1),SUMY1(1)     ANOV
     2         ,Q(1),A(MAXT1,1),TAUHAT(1),X1(10,1),SUMYJ(1)             ANOV
      DIMENSION SUMY1J(MAXT,1),BETHAT(1),P(1),B(MAXB1,1)                ANOV
      DIMENSION  W(1), V(1), WW(1), VV(1)                               ANOV
      DIMENSION IDEN(6)                                                 ANOV
      DIMENSION  THAT(1), BHAT(1)                                       ANOV
      REAL MNSSB                                                        ANOV
      REAL MNSST                                                        ANOV
      REAL MNSQB,MNSQBE                                                 ANOV
      REAL MNSQT, MNSQE                                                 ANOV
      DIMENSION INIT(37)                                                ANOV
      EQUIVALENCE (INIT(1),SUMSQX)                                      ANOV
      COMMON / AOV2NE /  IDEN,NAMET,NAMEB,IWT,SUMSQX,TAU,MNSQT,FRATIO,  ANOV
     2                   IERR,ERROR,MNSQE,SST,MNSST,FSST,INT,TIN,TINT,  ANOV
     3                   FINT,ITHT,WITHT,WIT,NSUM,SUMSQY,SUMSQT,BETA,   ANOV
     4                   MNSQB,FRATB,ERRORB,MNSQBE,SSE,MNSSB,FSSB,BIN,  ANOV
     5                   BINT,FINB,PFRAT,PFSST,PFINT,PFRATB,PFSSB,PFINB ANOV
      DO 4 I=1,37                                                       ANOV
    4 INIT(I)=0                                                         ANOV
 1008 FORMAT(////10X, A8,@ --- WEIGHTS@/(1X,6G20.7))                    ANOV
      GO TO (210,220,240),IWT                                           ANOV
```

```
  210 WT=1./FLOAT(ITOT)                                                  ANO\
C                                                                        ANO\
C         EQUAL WEIGHTS                                                  ANO\
C                                                                        ANO\
      DO 211 I=1,ITOT                                                    ANO\
  211 W(I)=WT                                                            ANO\
      WT=1./FLOAT(JTOT)                                                  ANO\
      DO 212 J=1,JTOT                                                    ANO\
  212 V(J)=WT                                                            ANO\
      GO TO 240                                                          ANO\
C                                                                        ANO\
C         PROPORTIONAL WEIGHTS                                           ANO\
C                                                                        ANO\
  220 SSS=0.                                                             ANO\
      DO 222 I=1,ITOT                                                    ANO\
      WT=0.                                                              ANO\
      DO 221 J=1,JTOT                                                    ANO\
  221 WT=WT+N(I,J)                                                       ANO\
      W(I)=1./WT                                                         ANO\
  222 SSS=SSS+W(I)                                                       ANO\
      SSS=1./SSS                                                         ANO\
      DO 225 I=1,ITOT                                                    ANO\
  225 W(I)=W(I)*SSS                                                      ANO\
      SSS=0.                                                             ANO\
      DO 224 J=1,JTOT                                                    ANO\
      WT=0.                                                              ANO\
      DO 223 I=1,ITOT                                                    ANO\
  223 WT=WT+N(I,J)                                                       ANO\
      V(J)=1./WT                                                         ANO\
  224 SSS=SSS+V(J)                                                       ANO\
      SSS=1./SSS                                                         ANO\
      DO 226 J=1,JTOT                                                    ANO\
  226 V(J)=V(J)*SSS                                                      ANO\
      GO TO 240                                                          ANO\
  240 WRITE(6,1008) NAMET,(W(I),I=1,ITOT)                                ANO\
      WRITE(6,1008) NAMED,(V(J),J=1,JTOT)                                ANO\
  250 NSUM=0                                                             ANO\
      SUMIJ=0.                                                           ANO\
      SUMSQY = 0.0                                                       ANO\
C                                                                        ANO\
C         FORM SUM OF SQUARES AND SUM OF N                               ANO\
C              AND SUM OF N AND OF Y OVER I                              ANO\
      DO 1 J = 1, JTOT                                                   ANO\
      NSUM1(J) = 0                                                       ANO\
      SUMYI(J) = 0.0                                                     ANO\
C                                                                        ANO\
      DO 1 I = 1, ITOT                                                   ANO\
      NSUM = NSUM + N(I, J)                                              ANO\
      NSUMI(J) = NSUM1(J) + N(I, J)                                      ANO\
      SUMYIJ(I,J)=0.                                                     ANO\
      K = N(I, J)                                                        ANO\
      DO 1  IK = 1, K                                                    ANO\
      SUMSQY = SUMSQY + Y(I, J, IK) * Y(I, J, IK)                        ANO\
      SUMYI(J) = SUMYI(J) + Y(I, J, IK)                                  ANO\
    1 CONTINUE                                                           ANO\
      IJTOT=ITOT*JTOT                                                    ANO\
      INT=IJTOT-ITOT-JTOT+1                                              ANO\
      ITHT=NSUM-IJTOT                                                    ANO\
      ITOTP1 = ITOT  + 1                                                 ANO\
      JTOTP1 = JTOT + 1                                                  ANO\
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      ITOT1=ITOT-1                                                      ANOV
      JTOT1=JTOT-1                                                      ANOV
      IERR=NSUM-ITOT-JTOT+1                                             ANOV
C                                                                       ANOV
C     CALCULATE AOV TABLE FOR ADJUSTED TAU EFFECTS                      ANOV
C                                                                       ANOV
C         FORM SUM OF N OVER J AND BEGIN TO FORM Q TO BE USED           ANOV
C         IN SOLUTION FOR TAU                                           ANOV
C                                                                       ANOV
      DO 2 I = 1, ITOT                                                  ANOV
      NSUMJ(I) = 0                                                      ANOV
      SUMYJ(I) = 0.0                                                    ANOV
      Q(I) = 0.0                                                        ANOV
      DO 2 J = 1, JTOT                                                  ANOV
      NSUMJ(I) = NSUMJ(I) + N(I, J)                                     ANOV
      Q(I)=Q(I) + FLOAT(N(I,J)) * SUMYI(J)/FLOAT(NSUMI(J))              ANOV
      K = N(I, J)                                                       ANOV
      DO 11  IK = 1, K                                                  ANOV
      SUMYIJ(I,J)=SUMYIJ(I,J) + Y(I,J,IK)                               ANOV
   11 SUMYJ(I) = SUMYJ(I) + Y(I, J, IK)                                 ANOV
      SUMIJ=SUMIJ + SUMYIJ(I,J)                                         ANOV
      SUMYIJ(I,J)=SUMYIJ(I,J)/FLOAT(K)                                  ANOV
    2 CONTINUE                                                          ANOV
      SUMIJ=SUMIJ/FLOAT(NSUM)                                           ANOV
      WRITE(6,3000) SUMIJ                                               ANOV
      WRITE(6,3003) NAMET                                               ANOV
      WRITE(6,3001) (SUMYJ(I),I=1,ITOT)                                 ANOV
      WRITE(6,3004) NAMET                                               ANOV
      WRITE(6,3005) (NSUMJ(I),I=1,ITOT)                                 ANOV
      WRITE(6,3003) NAMEB                                               ANOV
      WRITE(6,3001) (SUMYI(J),J=1,JTOT)                                 ANOV
      WRITE(6,3004) NAMEB                                               ANOV
      WRITE(6,3005) (NSUMI(J),J=1,JTOT)                                 ANOV
      WRITE(6,3002)                                                     ANOV
      DO 500 I=1,ITOT                                                   ANOV
      WRITE(6,3001) (SUMYIJ(I,J),J=1,JTOT)                              ANOV
  500 CONTINUE                                                          ANOV
      WRITE(6,3006)                                                     ANOV
      DO 501 I=1,ITOT                                                   ANOV
      WRITE(6,3005) (N(I,J),J=1,JTOT)                                   ANOV
  501 CONTINUE                                                          ANOV
 3000 FORMAT(@0MEAN EST @,G15.7)                                        ANOV
 3001 FORMAT(1X,10G12.4)                                                ANOV
 3002 FORMAT(@0CELL MEANS@)                                             ANOV
 3003 FORMAT(@0@, A8,@ SUMS@)                                           ANOV
 3004 FORMAT(@0@, A8,@ NUMBER SUMS@)                                    ANOV
 3005 FORMAT(1X,10I12)                                                  ANOV
 3006 FORMAT(@0CELL NUMBERS@)                                           ANOV
C                                                                       ANOV
C         CALCULATE WITHIN SUMS OF SQUARES                              ANOV
C                                                                       ANOV
      WITHT=0.                                                          ANOV
      DO 12 I=1,ITOT                                                    ANOV
      DO 12 J=1,JTOT                                                    ANOV
      K=N(I,J)                                                          ANOV
      DO 12 IK=1,K                                                      ANOV
   12 WITHT=WITHT + (Y(I,J,IK)-SUMYIJ(I,J))**2                          ANOV
      IF(ITHT.LE.0) GO TO 15                                            ANOV
      IF(WITHT.GT.0.) GO TO 112                                         ANOV
      WRITE(6,113) WITHT                                                ANOV
```

```
  113 FORMAT(@0 ***** WITHIN CELL ERROR IS @,G15.6,@ ... PROCESSING ABORANOV
     2TED ***** @)                                                      ANOV
      IWT=-1                                                            ANOV
      RETURN                                                            ANOV
  112 WT=WITHT/FLOAT(ITHT)                                              ANOV
C                                                                       ANOV
C         FORM THE MATRIX A AND AUGMENT IT TO ASTAR, A NONSINGULAR      ANOV
C         MATRIX, IN THE FOLLOWING MANNER                               ANOV
C                                                                       ANOV
C                              A(ITOT X ITOT)      1 (ITOT X 1)         ANOV
C                   ASTAR =                                             ANOV
C                              1-(1 X ITOT)        0 (1 X 1)           ANOV
C                                                                       ANOV
   15 DO 3 IR = 1, ITOT                                                 ANOV
      DO 3 IS = 1, ITOT                                                 ANOV
      A(IR, IS) = 0.0                                                   ANOV
      DO 5 J = 1, JTOT                                                  ANOV
    5 A(IR,IS) = A(IR,IS) + FLOAT(N(IR,J)*N(IS,J)) / FLOAT(NSUMI(J))    ANOV
      A(IR, IS) = -A(IR, IS)                                            ANOV
      IF(IR.EQ.IS) A(IR,IS)=FLOAT(NSUMJ(IR))+A(IR,IS)                   ANOV
    3 CONTINUE                                                          ANOV
C                                                                       ANOV
C                                                                       ANOV
      DO 7 I = 1, ITOT                                                  ANOV
      A(ITOTP1, I) = 1.0                                                ANOV
      A(I, ITOTP1) = 1.0                                                ANOV
C                                                                       ANOV
C         FINISH CALCULATION OF Q                                       ANOV
      Q(I) = SUMYJ(I) - Q(I)                                            ANOV
      X1(I,1)=Q(I)                                                      ANOV
    7 CONTINUE                                                          ANOV
C                                                                       ANOV
C         STORE 0 IN A(ITOTP1, ITOTP1)                                  ANOV
C                                                                       ANOV
      A(ITOTP1, ITOTP1) = 0.0                                           ANOV
C                                                                       ANOV
C         CALL MATRIX INVERSION SUBROUTINE ON A                         ANOV
C                                                                       ANOV
      CALL MATINV(A, ITOTP1, X1,  0, DET,MAXT1)                         ANOV
C                                                                       ANOV
C         CALCULATE TAU                                                 ANOV
C                                                                       ANOV
      DO 8 I = 1, ITOT                                                  ANOV
      TAUHAT(I) = 0.0                                                   ANOV
      DO 8 J = 1, ITOT                                                  ANOV
    8 TAUHAT(I) = TAUHAT(I) + A(I, J)*Q(J)                              ANOV
C                                                                       ANOV
      TAU = 0.0                                                         ANOV
      DO 9 I = 1, ITOT                                                  ANOV
    9 TAU = TAU + TAUHAT(I)* Q(I)                                       ANOV
C                                                                       ANOV
C         CALCULATE SUM OF SQUARES DUE TO MU AND BETA, UNADJUSTED       ANOV
C                                                                       ANOV
      SUMSQX = 0.0                                                      ANOV
      DO 10 J = 1, JTOT                                                 ANOV
   10 SUMSQX = SUMSQX + SUMYI(J) * SUMYI(J) /FLOAT(NSUMI(J))           ANOV
C                                                                       ANOV
C         AND MEAN SQUARE DUE TO TAU                                    ANOV
      MNSQT = TAU/FLOAT(ITOT1)                                          ANOV
C                                                                       ANOV
```

```
C          CALCULATE SUM OF SQUARES DUE TO ERROR                          ANOV
      ERROR = SUMSQY - SUMSQX - TAU                                       ANOV
      IF(ERROR.GT.0) GO TO 100                                            ANOV
      WRITE(6,101) ERROR                                                  ANOV
  101 FORMAT(@0 ***** ERROR SUM OF SQUARES IS @,G15.6,@ ... PROCESSING AANOV
     2BORTED ***** @)                                                     ANOV
      IWT=-2                                                              ANOV
      RETURN                                                              ANOV
  100 MNSQE=ERROR/FLOAT(IERR)                                             ANOV
C                                                                         ANOV
C          FROM WHICH AN F-RATIO WITH                                     ANOV
C                    (ITOT -1) AND (NSUM - JTOT -ITOT + 1)                ANOV
C          DEGREES OF FREEDOM                                             ANOV
C                                                                         ANOV
      FRATIO = MNSQT / MNSQE                                              ANOV
      PFRAT = FISH(FRATIO,ITOT1,IERR)                                     ANOV
      WRITE(6,2009)IDEN                                                   ANOV
      WRITE (6, 2002)                                                     ANOV
      WRITE(6,2004) NAMEB,JTOT,SUMSQX                                     ANOV
      WRITE(6,2005) NAMET,ITOT1,TAU,MNSQT,FRATIO,PFRAT                    ANOV
      WRITE (6, 2006) IERR, ERROR, MNSQE                                  ANOV
      SSS=0.                                                              ANOV
      SST=0.                                                              ANOV
      WT=0.                                                               ANOV
      DO 14 I=1,ITOT                                                      ANOV
      SSR=0.                                                              ANOV
      WW(I)=0.                                                            ANOV
      DO 13 J=1,JTOT                                                      ANOV
      SSR=SSR + V(J)*SUMYIJ(I,J)                                          ANOV
   13 WW(I)=WW(I) + V(J)*V(J)/FLOAT(N(I,J))                               ANOV
      WW(I)=1./WW(I)                                                      ANOV
      WT=WT + WW(I)                                                       ANOV
      SSS=SSS + WW(I)*SSR                                                 ANOV
      SST=SST + WW(I)*SSR*SSR                                             ANOV
   14 THAT(I)=SSR-SUMIJ                                                   ANOV
      SST=SST - SSS*SSS/WT                                                ANOV
      MNSST=SST/FLOAT(ITOT1)                                              ANOV
      IF(ITHT.LE.0) GO TO 16                                              ANOV
      FSST=MNSST/WIT                                                      ANOV
      PFSST = FISH(FSST,ITOT1,ITHT)                                       ANOV
   16 WRITE(6,2014) NAMET,ITOT1,SST,MNSST,FSST,PFSST                      ANOV
      TIN=ERROR-WITHT                                                     ANOV
      TINT=TIN/FLOAT(INT)                                                 ANOV
      IF(ITHT.LE.0) GO TO 17                                              ANOV
      FINT=TINT/WIT                                                       ANOV
      PFINT = FISH(FINT,INT,ITHT)                                         ANOV
   17 WRITE(6,2012) INT,TIN,TINT,FINT,PFINT                               ANOV
      WRITE(6,2013) ITHT,WITHT,WIT                                        ANOV
      WRITE (6, 2003) NSUM, SUMSQY                                        ANOV
      WRITE(6,2016) NAMET,(TAUHAT(I),I=1,ITOT)                            ANOV
      WRITE(6,2011) NAMET,(THAT(I),I=1,ITOT)                              ANOV
      WRITE(6,2010) NAMET,DET                                             ANOV
C                                                                         ANOV
C     CALCULATE AOV TABLE FOR ADJUSTED BETA EFFECTS                       ANOV
C                                                                         ANOV
      DO 53 IR= 1,JTOT                                                    ANOV
      DO 53 IS = 1,JTOT                                                   ANOV
      B(IR,IS) = 0.                                                       ANOV
      DO 55 J= 1, ITOT                                                    ANOV
   55 B(IR,IS) = B(IR,IS) + FLOAT(N(J,IR)*N(J,IS)) / FLOAT(NSUMJ(J))      ANOV
```

```
      B(IR,IS) = -B(IR,IS)                                              ANOV
      IF(IR.EQ.IS) B(IR,IS) = FLOAT(NSUMI(IR)) + B(IR,IS)               ANOV
   53 CONTINUE                                                          ANOV
      DO 52 J= 1,JTOT                                                   ANOV
      P(J) = 0.                                                         ANOV
      DO 52 I = 1,ITOT                                                  ANOV
      P(J) = P(J) + FLOAT(N(I,J)) * SUMYJ(I)/FLOAT(NSUMJ(I))            ANOV
   52 CONTINUE                                                          ANOV
      DO 57 J=1,JTOT                                                    ANOV
      B(JTOTP1,J) = 1.                                                  ANOV
      B(J,JTOTP1) = 1.                                                  ANOV
C                                                                       ANOV
C          FINISH CALCULATION OF P                                      ANOV
      P(J) = SUMYI(J) - P(J)                                            ANOV
      X1(J,1)=P(J)                                                      ANOV
   57 CONTINUE                                                          ANOV
      B(JTOTP1,JTOTP1) = 0.                                             ANOV
C                                                                       ANOV
C        CALL MATRIX INVERSION SUBROUTINE ON B                          ANOV
C                                                                       ANOV
      CALL MATINV(B,JTOTP1,X1,0,DET,MAXB1)                              ANOV
C                                                                       ANOV
C        CALCULATE BETA                                                 ANOV
C                                                                       ANOV
      DO 58 I = 1,JTOT                                                  ANOV
      BETHAT(I) = 0.                                                    ANOV
      DO 58 J = 1,JTOT                                                  ANOV
   58 BETHAT(I) = BETHAT(I) + B(I,J)*P(J)                               ANOV
C                                                                       ANOV
      BETA = 0.                                                         ANOV
      DO 59 J = 1,JTOT                                                  ANOV
   59 BETA = BETA + BETHAT(J)*P(J)                                      ANOV
C                                                                       ANOV
C        CALCULATE SUM OF SQUARES DUE TO MU AND TAU,UNADJUSTED          ANOV
C                                                                       ANOV
      SUMSQT = 0.                                                       ANOV
      DO 60 I = 1,ITOT                                                  ANOV
   60 SUMSQT = SUMSQT + SUMYJ(I) * SUMYJ(I) /FLOAT(NSUMJ(I))           ANOV
C                                                                       ANOV
C        CALCULATE SUM OF SQUARES DUE TO ERROR                          ANOV
      ERRORB = SUMSQY - SUMSQT - BETA                                   ANOV
C                                                                       ANOV
C        AND MEAN SQUARE DUE TO BETA                                    ANOV
      MNSQB = BETA/FLOAT(JTOT1)                                         ANOV
C                                                                       ANOV
C        AND DUE TO ERROR                                               ANOV
      IF(ERRORB.GT.0.) GO TO 160                                        ANOV
      WRITE(6,101) ERRORB                                               ANOV
      IWT=-3                                                            ANOV
      RETURN                                                            ANOV
  160 MNSQPE=ERRORB/FLOAT(IERR)                                         ANOV
C                                                                       ANOV
C        AND F-RATIO                                                    ANOV
      FRATB = MNSQB  / MNSQPE                                           ANOV
      PFRATB = FISH(FRATB,JTOT1,IERR)                                   ANOV
      WRITE(6,2009)IDEN                                                 ANOV
      WRITE(6,2002)                                                     ANOV
      WRITE(6,2004) NAMET,ITOT,SUMSQT                                   ANOV
      WRITE(6,2005) NAMEB,JTOT1,BETA,MNSQB,FRATB,PFRATB                 ANOV
      WRITE(6,2006) IERR,ERRORB,MNSQBE                                  ANOV
```

```
      WT=0.                                                             ANOV
      SSB=0.                                                            ANOV
      SSS=0.                                                            ANOV
      DO 64 J=1,JTOT                                                    ANOV
      SSR=0.                                                            ANOV
      VV(J)=0.                                                          ANOV
      DO 63 I=1,ITOT                                                    ANOV
      SSR=SSR + SUMYIJ(I,J)*W(I)                                        ANOV
   63 VV(J)=VV(J) + W(I)*W(I)/FLOAT(N(I,J))                             ANOV
      VV(J)=1./VV(J)                                                    ANOV
      WT=WT + VV(J)                                                     ANOV
      SSS=SSS + VV(J)*SSR                                               ANOV
      SSB=SSB + VV(J)*SSR*SSR                                           ANOV
   64 BHAT(J)=SSR-SUMIJ                                                 ANOV
      SSB=SSB-SSS*SSS/WT                                                ANOV
      MNSSB=SSB/FLOAT(JTOT1)                                            ANOV
      IF(ITHT.LE.0) GO TO 66                                            ANOV
      FSSB=MNSSB/WIT                                                    ANOV
      PFSSB = FISH(FSSB,JTOT1,ITHT)                                     ANOV
   66 WRITE(6,2014) NAMEB,JTOT1,SSB,MNSSB,FSSB,PFSSB                    ANOV
      BIN=ERRORB-WITHT                                                  ANOV
      BINT=BIN/FLOAT(INT)                                               ANOV
      IF(ITHT.LE.0) GO TO 65                                            ANOV
      FINB=BINT/WIT                                                     ANOV
      PFINB = FISH(FINB,INT,ITHT)                                       ANOV
   65 WRITE(6,2012) INT,BIN,BINT,FINB,PFINB                             ANOV
      WRITE(6,2013) ITHT,WITHT,WIT                                      ANOV
      WRITE(6,2003) NSUM,SUMSQY                                         ANOV
      WRITE(6,2016) NAMEB,(BETHAT(J),J=1,JTOT)                          ANOV
      WRITE(6,2011) NAMEB,(BHAT(J),J=1,JTOT)                            ANOV
      WRITE(6,2010) NAMEB,DET                                           ANOV
      RETURN                                                            ANOV
 2002 FORMAT(@0@,40X,@UNBALANCED TWO-WAY ANALYSIS OF VARIANCE@///       ANOV
     21H , 18X, 2HSV, 19X, 2HDF, 19X, 2HSS, 19X, 2HMS, 20X, 1HF,       ANOV
     317X,@PROB F@)                                                     ANOV
 2003 FORMAT (1H0, 10X, 5HTOTAL, 20X, I5, 15X, G15.7)                   ANOV
 2004 FORMAT(1H0,10X,@MU@, @, A8,13X,I5,15X,G15.7/11X,@(UNADJUSTED)@)   ANOV
 2005 FORMAT(1H0,10X, A8,@ (ADJ.)@,10X,I5,15X,3(G15.7,5X),G15.7         ANOV
     2       /11X,@ASSUME ADDITIVE@)                                    ANOV
 2006 FORMAT(1H0,10X,@ERROR@,20X,I5,15X,2(G15.7,5X)/11X,@ADDITIVE@)     ANOV
 2009 FORMAT(1H1 /////////// 30X,8A10)                                   ANOV
 2010 FORMAT (/////,1X,A8,@ EFFECT MATRIX DETERMINANT = @,G15.7)        ANOV
 2011 FORMAT(////10X,@ESTIMATES OF @, A8,@  EFFECTS@/(1X,6G20.7))       ANOV
 2012 FORMAT(1H0,10X,@INTERACTION@,14X,I5,15X,3(G15.7,5X),G15.7)        ANOV
 2013 FORMAT(1H0,10X,@WITHIN CELL ERROR@,8X,I5,15X,2(G15.7,5X))         ANOV
 2014 FORMAT(1H0,10X, A8,@(ADJ.)@,11X,I5,15X,3(G15.7,5X),G15.7          ANOV
     2       /11X,@(WEIGHTED)@)                                         ANOV
 2016 FORMAT(////10X,@ESTIMATES OF @, A8,@  EFFECTS ASSUMING NO INTERACTANOV
     2ION@/(1X,6G20.7))                                                 ANOV
      END                                                               ANOV
      SUBROUTINE MATINV(A,N,F,M,DETERM,MAX)                             MATV
      DIMENSION IPIVOT(21), A(MAX, 10), B(10, 1), INDEX(21, 2),         MATV
     2          PIVOT(21)                                               MATV
      EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM), (AMAX, T, SWAP)         MATV
C     INITIALIZATION                                                    MATV
   10 DETERM=1.0                                                        MATV
   15 DO 20 J=1,N                                                       MATV
   20 IPIVOT(J)=0                                                       MATV
   30 DO 550 I=1,N                                                      MATV
C     SEARCH FOR PIVOT ELEMENT                                          MATV
```

```
   40 AMAX=0.0                                                          MATV
   45 DO 105 J=1,N                                                      MATV
   50 IF (IPIVOT(J)-1) 60, 105, 60                                      MATV
   60 DO 100 K=1,N                                                      MATV
   70 IF (IPIVOT(K)-1) 80, 100, 740                                     MATV
   80 IF(ABS(AMAX)-ABS(A(J,K)))85,100,100                               MATV
   85 IROW=J                                                            MATV
   90 ICOLUM=K                                                          MATV
   95 AMAX=A(J,K)                                                       MATV
  100 CONTINUE                                                          MATV
  105 CONTINUE                                                          MATV
  110 IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                                   MATV
C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL                 MATV
  130 IF (IROW-ICOLUM) 140, 260, 140                                    MATV
  140 DETERM=-DETERM                                                    MATV
  150 DO 200 L=1,N                                                      MATV
  160 SWAP=A(IROW,L)                                                    MATV
  170 A(IROW,L)=A(ICOLUM,L)                                             MATV
  200 A(ICOLUM,L)=SWAP                                                  MATV
  205 IF(M) 260, 260, 210                                               MATV
  210 DO 250 L=1, M                                                     MATV
  220 SWAP=B(IROW,L)                                                    MATV
  230 B(IROW,L)=B(ICOLUM,L)                                             MATV
  250 B(ICOLUM,L)=SWAP                                                  MATV
  260 INDEX(I,1)=IROW                                                   MATV
  270 INDEX(I,2)=ICOLUM                                                 MATV
  310 PIVOT(I)=A(ICOLUM,ICOLUM)                                         MATV
  320 DETERM=DETERM*PIVOT(I)                                            MATV
C     DIVIDE PIVOT ROW BY PIVOT ELEMENT                                 MATV
  330 A(ICOLUM,ICOLUM)=1.0                                              MATV
  340 DO 350 L=1,N                                                      MATV
  350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)                                  MATV
  355 IF(M) 380, 380, 360                                               MATV
  360 DO 370 L=1,M                                                      MATV
  370 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT(I)                                  MATV
C     REDUCE NON-PIVOT ROWS                                             MATV
  380 DO 550 L1=1,N                                                     MATV
  390 IF(L1-ICOLUM) 400, 550, 400                                       MATV
  400 T=A(L1,ICOLUM)                                                    MATV
  420 A(L1,ICOLUM)=0.0                                                  MATV
  430 DO 450 L=1,N                                                      MATV
  450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T                                     MATV
  455 IF(M) 550, 550, 460                                               MATV
  460 DO 500 L=1,M                                                      MATV
  500 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T                                     MATV
  550 CONTINUE                                                          MATV
C     INTERCHANGE COLUMNS                                               MATV
  600 DO 710 I=1,N                                                      MATV
  610 L=N+1-I                                                           MATV
  620 IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630                          MATV
  630 JROW = INDEX(L,1)                                                 MATV
  640 JCOLUM=INDEX(L,2)                                                 MATV
  650 DO 705 K=1,N                                                      MATV
  660 SWAP=A(K,JROW)                                                    MATV
  670 A(K,JROW)=A(K,JCOLUM)                                             MATV
  700 A(K,JCOLUM)=SWAP                                                  MATV
  705 CONTINUE                                                          MATV
  710 CONTINUE                                                          MATV
  740 RETURN                                                            MATV
      END                                                               MATV
```

```
      FUNCTION TWOAVT(EMS,JSUB,N2,ALPHA,W,II,IS)                        FUNCTS
      COMMON DUM(357),Y(729),A(4,4),B(4,4)                              FUNCTS
      TWOAVT=SQRT(EMS*JSUB*FISHIN(ALPHA,JSUB,N2)*(A(II,II)-2.*A(II,IS)  FUNCTS
     2            +A(IS,IS)))                                           FUNCTS
      RETURN                                                            FUNCTS
      END                                                               FUNCTS
      FUNCTION TWOAVB(EMS,JSUB,N2,ALPHA,W,II,IS)                        FUNCTS
      COMMON DUM(357),Y(729),A(4,4),B(4,4)                              FUNCTS
      TWOAVB=SQRT(EMS*JSUB*FISHIN(ALPHA,JSUB,N2)*(B(II,II)-2.*B(II,IS)  FUNCTS
     2            +B(IS,IS)))                                           FUNCTS
      RETURN                                                            FUNCTS
      END                                                               FUNCTS
      FUNCTION TWOWT (EMS,JSUB,N2,ALPHA,W,II,IS)                        FUNCTS
      DIMENSION W(1)                                                    FUNCTS
      TWOWT=SQRT(EMS*JSUB*FISHIN(ALPHA,JSUB,N2)*(1./W(II) + 1./W(IS)))  FUNCTS
      RETURN                                                            FUNCTS
      END                                                               FUNCTS
      SUBROUTINE SCHEFE(K,TMEANS,EMS,N2,ALPHA,KOE,W,FDUM)               SCHEFE
C                                                                       SCHEFE
C     SCHEFFE'S CONTRASTS FOR SIGNIFICANT DIFFERENCES IN MEANS          SCHEFE
C                                                                       SCHEFE
C  K=NUMBER OF MEANS TO BE TESTED   TMEANS=VECTOR OF MEANS TO BE TESTED SCHEFE
C  EMS=ERROR MEAN SQUARE (DENOMINATOR OF F TEST)                        SCHEFE
C  N2 =ERROR DEGREES OF FREEDOM                                         SCHEFE
C     FDUM IS A DUMMY EXTERNALLY DEFINED FUNCTION TO CALCULATE CONTRAST SCHEFE
C FOR STANDARD OUTPUT (K.LE.10) ALL MEANS HAVING THE SAME LINE UNDER THESCHEFE
C     STATISTICALLY THE SAME.                                           SCHEFE
      DIMENSION TMEANS(K),PRINT(20),KOE(1),W(1)                         SCHEFE
      DATA BLANK /6H      /,XLINE / 6HXXXXXX /                          SCHEFE
      ISAVE2=0                                                          SCHEFE
C RANK MEANS FROM LOW TO HIGH                                           SCHEFE
      CALL ORDERM(TMEANS,K,KOE)                                         SCHEFE
      WRITE(6,51) (KOE(I),I=1,K)                                        SCHEFE
   51 FORMAT(' ',10(6X,I2,4X))                                          SCHEFE
      WRITE(6,53)(TMEANS(I),I=1,K)                                      SCHEFE
 53   FORMAT(1X ,10G12.6)                                               SCHEFE
C DO COMPARISONS                                                        SCHEFE
      ISTOP=K                                                           SCHEFE
 25   ISM1=ISTOP-1                                                      SCHEFE
      DO 24 I=1,20                                                      SCHEFE
 24   PRINT(I)=BLANK                                                    SCHEFE
      DO 30 I=1,ISM1                                                    SCHEFE
      ISAVE=I                                                           SCHEFE
      JSUB=ISM1-I+1                                                     SCHEFE
 29   IF(ISAVE2.EQ.I    )GO TO 31                                       SCHEFE
      RANGE = TMEANS(ISTOP)-TMEANS(I)                                   SCHEFE
      II=KOE(I)                                                         SCHEFE
      IS=KOE(ISTOP)                                                     SCHEFE
      S=FDUM(EMS,JSUB,N2,ALPHA,W,II,IS)                                 SCHEFE
      IF(RANGE-S) 34,34,30                                              SCHEFE
 30   CONTINUE                                                          SCHEFE
 31   ISTOP=ISTOP-1                                                     SCHEFE
      IF(ISTOP.EQ.1)GO TO 99                                            SCHEFE
      GO TO 25                                                          SCHEFE
 34   IF(K.LE.10)GO TO 35                                               SCHEFE
      WRITE(6,33)ALPHA,(TMEANS(KKK),KKK=ISAVE,ISTOP)                    SCHEFE
 33   FORMAT( '0 THE FOLLOWING MEANS ARE STATISTICALLY THE SAME AT THE 'SCHEFE
     2      ,F5.3,' ALPHA LEVEL',/,(10G12.6))                          SCHEFE
      GO TO 39                                                          SCHEFE
 35   DO 37 I=ISAVE,ISTOP                                               SCHEFE
```

```
      II=2*I                                                            SCHEFE
      PRINT(II-1) = XLINE                                               SCHEFE
   37 PRINT(II) = XLINE                                                 SCHEFE
      WRITE(6,52)(PRINT(KK),KK=1,20)                                    SCHEFE
 38   IF(ISAVE.EQ.1)GO TO 99                                            SCHEFE
      ISAVE2=ISAVE                                                      SCHEFE
      GO TO 31                                                          SCHEFE
 99   RETURN                                                            SCHEFE
 52   FORMAT(@ @,20A6)                                                  SCHEFE
      END                                                               SCHEFE
       SUBROUTINE ORDERM(X,N,KOE)                                       ORDER
       DIMENSION X(N)   ,KOE(N)                                         ORDER
      NN=N                                                              ORDER
       K1=2                                                             ORDER
      DO 80 I=1,NN                                                      ORDER
   80 KOE(I)=I                                                          ORDER
    4  DO 99 I=K1,NN                                                    ORDER
       IF(X(I).LT.X(I-1)) GOTO 76                                       ORDER
   99 CONTINUE                                                          ORDER
      RETURN                                                            ORDER
   76  DO 82 K= 1,NN                                                    ORDER
      IF(X(I).LT.X(K)) GO TO 84                                         ORDER
   82 CONTINUE                                                          ORDER
   84 Z=X(I)                                                            ORDER
      KK=KOE(I)                                                         ORDER
      I1=I-1                                                            ORDER
      DO 86KJ=K,I1                                                      ORDER
      J=K+I1-KJ                                                         ORDER
      KOE(J+1)=KOE(J)                                                   ORDER
   86 X(J+1)=X(J)                                                       ORDER
      X(K)=Z                                                            ORDER
      KOE(K)=KK                                                         ORDER
      K1=I+1                                                            ORDER
       IF(K1.GT.NN) RETURN                                              ORDER
       GOTO 4                                                           ORDER
       END                                                              ORDER
       FUNCTION FISHIN(ALPHA,N1,N2)                                     FISHIN
C-------------------------------------------------------------------------FISHIN
C CALCULATES THE INVERSE @F@ VALUE GIVEN THE CONFIDENCE COEFFICIENT      FISHIN
C   ALPHA AND THE DEGREES OF FREEDOM(N).                                FISHIN
C-------------------------------------------------------------------------FISHIN
      Y1=N1                                                             FISHIN
      Y2=N2                                                             FISHIN
C-------------------------------------------------------------------------FISHIN
C  ADJUST FOR DEGREES OF FREEDOM EQUAL TO 1                             FISHIN
C-------------------------------------------------------------------------FISHIN
      IF(N1.EQ.1)  Y1=2                                                 FISHIN
      IF(N2.EQ.1)  Y2=2                                                 FISHIN
C-------------------------------------------------------------------------FISHIN
C CALL PHINV TO GET INVERSE NORMAL VALUE OF 1.-ALPHA                    FISHIN
C-------------------------------------------------------------------------FISHIN
       X=PHINV(1.-ALPHA)                                                FISHIN
C-------------------------------------------------------------------------FISHIN
C  COMPUTE LAMDA VALUE                                                  FISHIN
C-------------------------------------------------------------------------FISHIN
      Y=(X**2-3.)/6.                                                    FISHIN
      IC=0                                                              FISHIN
C-------------------------------------------------------------------------FISHIN
C  COMPUTE THE INITIAL APPROXIMATION TO THE INVERSE @F@ FUNCTION        FISHIN
C-------------------------------------------------------------------------FISHIN
```

```
      Y1=1./(Y1-1.)                                                     FISHIN
      Y2=1./(Y2-1.)                                                     FISHIN
      H=2./(Y1+Y2)                                                      FISHIN
      X=X*SQRT(H+Y)/H-(Y1-Y2)*(Y+5./6.-2./(3.*H))                       FISHIN
      X=EXP(2.*X)                                                       FISHIN
C-----------------------------------------------------------------------FISHIN
C  COMPUTE THE CONSTANT TO THE aFa DISTRIBUTION,TESTING FOR N1 AND/OR N2FISHIN
C   ODD OR EVEN.                                                        FISHIN
C-----------------------------------------------------------------------FISHIN
      G=1.                                                              FISHIN
      IB1=2                                                             FISHIN
      IF(MOD(N1,2).EQ.0) GO TO 1                                        FISHIN
      G=1.7724539                                                       FISHIN
      IB1=1                                                             FISHIN
    1 IB2=2                                                             FISHIN
      IF(MOD(N2,2).EQ.0) GO TO 2                                        FISHIN
      G=G*1.7724539                                                     FISHIN
      IB2=1                                                             FISHIN
    2 IB3=2                                                             FISHIN
      IF(MOD(N1+N2,2).EQ.0) GO TO 3                                     FISHIN
      G=G/1.7724539                                                     FISHIN
      IB3=1                                                             FISHIN
    3 IF((IB1+IB2).NE.2)  G=2.*G                                        FISHIN
      IF((N1+N2).LE.3)  GO TO 5                                         FISHIN
      ND=N1+N2-2-IB3                                                    FISHIN
      ND1 = ND + 1                                                      FISHIN
      DO 4 II=1,ND1,2                                                   FISHIN
      I = II - 1                                                        FISHIN
      IF((IB1+I).LE.(N1-2))  G=G*(IB1+I)                                FISHIN
      IF((IB2+I).LE.(N2-2))  G=G*(IB2+I)                                FISHIN
    4 G=G/(IB3+I)                                                       FISHIN
C-----------------------------------------------------------------------FISHIN
C  COMPUTE THE VALUE OF FISHIN                                          FISHIN
C-----------------------------------------------------------------------FISHIN
    5 Y2=N2/(N2+N1*X)                                                   FISHIN
      Y1=1.-Y2                                                          FISHIN
      Y=1.+(G*(1.-ALPHA-FISH(X,N1,N2)))/SQRT(Y1**N1*Y2**N2)             FISHIN
      FISHIN=X*Y                                                        FISHIN
C-----------------------------------------------------------------------FISHIN
C  IF FISHIN IS NEGATIVE, RESET FISHIN TO .5*LAST APPROXIMATION(X).     FISHIN
C-----------------------------------------------------------------------FISHIN
      IF(Y.LT.0.)  FISHIN=.5*X                                          FISHIN
C-----------------------------------------------------------------------FISHIN
C  IF THE ABSOLUTE VALUE OF THE DIFFERENCE IS LESS THAN .5E-6, RETURN   FISHIN
C-----------------------------------------------------------------------FISHIN
      IF(ABS(X/FISHIN-1.).LT.(.5E-6))  GO TO 7                          FISHIN
C-----------------------------------------------------------------------FISHIN
C  IF THE RELATIVE VALUE OF THE DIFFERENCE IS LESS THAN .5E-6, RETURN   FISHIN
C-----------------------------------------------------------------------FISHIN
      IF(ABS(X-FISHIN).LT.(.5E-6)) GO TO 7                              FISHIN
      IC=IC+1                                                           FISHIN
      IF(IC.GT.100)  RETURN                                             FISHIN
C-----------------------------------------------------------------------FISHIN
C  SET THE APPROXIMATION EQUAL TO FISHIN AND CONTINUE TO ITERATE        FISHIN
C-----------------------------------------------------------------------FISHIN
      X=FISHIN                                                          FISHIN
      GO TO 5                                                           FISHIN
    7 RETURN                                                            FISHIN
      END                                                               FISHIN
      FUNCTION PHINV( P )                                               PHINV
```

```
      IF(P .EQ. 1.0)GO TO 98                                                    PHINV
      IF(P .EQ. 0.0)GO TO 97                                                    PHINV
      IF(P .GT. 1.0)GO TO 88                                                    PHINV
      IF(P .LT. 0.0)GO TO 88                                                    PHINV
      K = 1                                                                     PHINV
      IF(P .GT. 0.5)GO TO 47                                                    PHINV
    8 T3=SQRT(-2.0*ALOG(P))                                                     PHINV
      T4P=2.515517+.802853*T3+.010328*T3*T3                                     PHINV
      T5P=1.0+1.432788*T3+.189269*T3*T3+.001308*T3*T3*T3                        PHINV
      XT=T3-T4P/T5P                                                             PHINV
      XT=-XT                                                                    PHINV
   13 DO 53 I=1,100                                                             PHINV
      PHP = EXP(-0.5*XT*XT)                                                     PHINV
      PT = PHI (XT)                                                             PHINV
      IF(ABS(P-PT) .LT. P*4.0E-8)GO TO 99                                       PHINV
      Z = (P-PT)*2.50662827 / PHP                                               PHINV
      XT = XT + Z                                                               PHINV
   53 CONTINUE                                                                  PHINV
      GO TO 99                                                                  PHINV
   47  P = 1.0 - P                                                             PHINV
      K = 2                                                                     PHINV
      GO TO 8                                                                   PHINV
   99 GO TO (26,27),K                                                           PHINV
   26 PHINV = XT                                                                PHINV
      RETURN                                                                    PHINV
   27 PHINV = -XT                                                               PHINV
      P = 1.0 - P                                                               PHINV
      RETURN                                                                    PHINV
   98 PHINV = 1.0E+38                                                           PHINV
      RETURN                                                                    PHINV
   97 PHINV = -1.0E+38                                                          PHINV
      RETURN                                                                    PHINV
   88 WRITE(6,10) P                                                             PHINV
   10 FORMAT(1H0,5X,29HARGUMENT NOT A PROBABILITY = ,5X,E14.7  )                PHINV
      RETURN                                                                    PHINV
      END                                                                       PHINV
```

D.6 Program RETD as modified to perform the basic analysis plus an analysis of variance

```
      PROGRAM RETD(INPUT,OUTPUT,TAPE6=OUTPUT)                          FIX
C     MAIN PROGRAM                                                     RETD
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP       XREDUC
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(246),INDX3(124)         RETD
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVCL(150)          XRED
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),  RETD
     *                LTYPE(20),ITP,LTEST(125),ITT,NSMF,NDATA          RETD
      COMMON/RG/ DATESC(3,5),IVV(4), SSMEAN(3), SSIGMA(3), NNN(3),SSE(3)RETD
      COMMON /SISBUF/ ILOF(35),IBL(125),IOR(35),IBUFF(30),KEY,KYE,     RETD
     +                MFUNC,IFILE,IERR                                 RETD
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)          RETD
C     START,LAUNCH,SPLASH DOWN                                         JAN2
      DATA DATESC/180.,208.,265.,114.,144.,172.,189.,209.,268.,        DEC1
     X 285.,320.,404.,  290.,342.,354./                                DEC1
C     SL4 SPLASH DOWN IS 39  FEB 8 1974                                DEC1
      IOPLOT = 0                                                       RETD
      KSKIP = 0                                                        RETD
C     IFFF IS LOOP CONTROL TO ALLOW OVERLAY                            JAN2
      IFFF = -1                                                        RETD
    1 CALL RET                                                         RETD
      CALL STAT                                                        RETD
      GO TO 1                                                          RETD
      END                                                              RETD
      SUBROUTINE LINE(X,Y,N,M,I,LL)                                    LINE
C     X IS THE DATE. (INDEPENDENT VARIABLE)                            LINE
C     Y IS THE DATA. (DEPENDENT VARIABLE)                              LINE
C     IFLAG = 13   LAST POINT NOT VALID                                LINE
C     JFLAG = 13   LAST POINT VALID                                    LINE
C     ISFLAG IS THE NUMBER OF POINTS PLOTED.                           LINE
      DIMENSION X(1), Y(1)                                             LINE
      DATA D/0.04/                                                     LINE
      IFLAG = 0                                                        LINE
      JFLAG = 0                                                        LINE
      ISFLAG = 0                                                       LINE
      IP = 3                                                           LINE
      IF(N - 1) 9,1,1                                                  LINE
    1 DO 6 I=1,N                                                       LINE
      II = I*M - M + 1                                                 LINE
      XP = (X(II) - X(N + M))/X(N + 2*M)                               LINE
      YP = (Y(II) - Y(N + M))/Y(N + 2*M)                               LINE
C     CHECK TO SEE IF POINT VALUE IS VALID.                            LINE
      IF(Y(II)) 3,13,23                                                LINE
    3 IY = -Y(II) + 0.5                                                LINE
      IF(IY - 2) 13,14,13                                              LINE
C     THIS IS ZERO ON THE GRAPH.                                       LINE
   14 YP = -Y(N+1)/Y(N+2*M)                                            LINE
      GO TO 23                                                         LINE
C     INVALID POINT.                                                   LINE
   13 IFLAG = 13                                                       LINE
      GO TO 6                                                          LINE
C     VALID POINT.  CHECK FOR CONTINUOUS DATE.                         LINE
   23 JFLAG = 13                                                       LINE
      IF(ISFLAG) 25,25,24                                              LINE
   24 IF(X(II) - X(II-M) - 1.5) 25,35,35                               LINE
   25 IF(IFLAG) 35,26,35                                               LINE
C     PLOT THIS PEN DOWN.                                              LIN
   26 CALL PLOT(XP,YP,IP)                                              LINE
      ISFLAG = ISFLAG + 1                                              LINE
      IP = 2                                                           LINE
      GO TO 6                                                          LINE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C     DISCONTINUOUS GRAPH, PLOT AN X ON EACH END OF LINE.                    LINE
  35  CALL WHERE(U,V,F)                                                      LINE
      IFLAG = 0                                                              LINE
      KFLAG = JFLAG*ISFLAG                                                   LINE
      IF(KFLAG) 16,17,16                                                     LINE
  16  CALL SYMBOL(U,V,D,LL,0.0,-1)                                           LINE
      ISFLAG = ISFLAG + 1                                                    LINE
  17  CALL SYMBOL(XP,YP,D,LI,0.0,-1)                                         LINE
      IP = 2                                                                 LINE
      ISFLAG = ISFLAG + 1                                                    LINE
   6  CONTINUE                                                               LINE
      KFLAG = IFLAG*JFLAG                                                    LINE
      IF(KFLAG) 5,9,29                                                       LINE
  29  CALL WHERE(U,V,F)                                                      LINE
      CALL SYMBOL(U,V,D,LL,0.0,-1)                                           LINE
      ISFLAG = ISFLAG + 1                                                    LINE
   9  RETURN                                                                 LINE
      END                                                                    LINE
      SUBROUTINE AXIS (X,Y,BCD,NC,SIZE,THETA,YMIN,DY,NDEC,NLAB,NTIC)         AXIS
      DIMENSION D(2), H(11)                                                  AXIS
      DATA D/.8,.56/                                                         AXIS
      DATA H/.56,.4,.28,.2,.14,.1,.07,.05,.035,.025,.0175/                   AXIS
      AC = NC                                                                AXIS
      SIG=SIGN(1.0,AC)                                                       AXIS
   2  NAC=IABS(NC)                                                           AXIS
      TH=THETA*0.017453294                                                   AXIS
      IF(NLAB.LE.0) NLAB = 1                                                 AXIS
      IF(NTIC.LE.0) NTIC = 1                                                 AXIS
      FNLAB = NLAB                                                           AXIS
      M = SIZE*FNLAB + 0.1                                                   AXIS
C     M = SIZE + 0.50                                                        AXIS
      CTH = COS  (TH)                                                        AXIS
      STH = SIN  (TH)                                                        AXIS
      CTL = CTH/FNLAB                                                        AXIS
      STL = STH/FNLAB                                                        AXIS
      TH = F                                                                 AXIS
      N1 = I  + 1                                                            AXIS
      N2 = NJ/2                                                              AXIS
      ADY=ABS(DY/FNLAB)                                                      AXIS
C     ADY=ABS(DY)                                                            AXIS
      STAT=YMIN                                                              AXIS
      EXP = 0.0                                                              AXIS
      IF ( ADY ) 9,18,9                                                      AXIS
   9  IF ( ADY -100.0 ) 10,12,12                                             AXIS
  12  ADY = ADY / 10.0                                                       AXIS
      STAT=STAT/10.0                                                         AXIS
      EXP = EXP + 1.0                                                        AXIS
      GO TO 9                                                                AXIS
  14  ADY = ADY * 10.0                                                       AXIS
      STAT=STAT*10.0                                                         AXIS
      EXP = EXP - 1.0                                                        AXIS
  10  IF ( ADY - 1.00 ) 14,18,18                                             AXIS
C 10  IF ( ADY - 0.01 ) 14,18,18                                             AXIS
C 18  XA = X   - (.20 * SIG   - .05) *STH - .0857 * CTH                      AXIS
  18  XA = X   - (H(NLAB+1)  * SIG   - H(NLAB+5)) *STH - .0857 * CTH        AXIS
C     YA = Y   + (.20 * SIG   - .05) * CTH - .0857 * STH                     AXIS
      YA = Y   + (H(NLAB+1) * SIG   - H(NLAB+5)) * CTH - .0857 * STH         AXIS
      I = 0                                                                  AXIS
  25  I = I + 1                                                              AXIS
C     CALL NUMBER (XA,YA,0.1,STAT,THETA,2)                                   AXIS
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      CALL NUMBER (XA,YA,H(NLAB+3),STAT,THETA,NDEC)                    AXIS
      STAT=STAT+SIGN(ADY,DY)                                           AXIS
C     XA = XA + CTH                                                    AXIS
      XA = XA + CTN                                                    AXIS
C     YA = YA + STH                                                    AXIS
      YA = YA + STN                                                    AXIS
      IF(I - N2) 25,31,26                                              AXIS
   26 IF(I - N1) 25,60,60                                              AXIS
   31 TNC = NAC + 7                                                    AXIS
C     XC = X + (SIZE / 2.0 -.06 * TNC)*CTH - (-.07 * SIG *.36)* STH    AXIS
      XC = X + (SIZE / 2.0 -H(NLAB+4) * TNC)*CTH                       AXIS
     1- (-H(NLAB+4) + SIG *(H(NLAB) + H(NLAB+3)))* STH                 AXIS
C     YC = Y + (SIZE / 2.0 -.06 * TNC)*STH + (-.07 + SIG *.36)* CTH    AXIS
      YC = Y + (SIZE / 2.0 -H(NLAB+4) * TNC)*STH                       AXIS
     1+ (-H(NLAB+4) + SIG *(H(NLAB) + H(NLAB+3)))* CTH                 AXIS
C     CALL SYMBOL (XC,YC,0.14,BCD,THETA,NAC)                           AXIS
      CALL SYMBOL (XC,YC,H(NLAB+2),BCD,THETA,NAC)                      AXIS
      XC = XC +((TNC -6.0) * 0.12)* CTH                                AXIS
      YC = YC +((TNC -6.0) * 0.12)* STH                                AXIS
      IF (EXP) 35,50,35                                                AXIS
C  35 CALL SYMBOL (XC,YC,0.14,6(X10  )@ ,THETA,7)                      AXIS
C  35 CALL SYMBOL (XC,YC,H(NLAB+2),@(X10  )@ ,THETA,7)                 AXIS
   35 CALL SYMBOL (XC,YC,H(NLAB+2),7H(X10  ),THETA,7)                  AXIS
C     XC = XC + .48 * CTH  -.07 * STH                                  AXIS
      XC=XC+.38*CTH-H(NLAB+4)*STH                                      AXIS
C     YC = YC + .48 * STH  +.07 * CTH                                  DEC10
      YC=YC+.38*STH+H(NLAB+4)*CTH                                      AXIS
C  40 CALL NUMBER (XC,YC,0.10,EXP,THETA,-1)                            DEC10
   40 CALL NUMBER (XC,YC,H(NLAB+3),EXP,THETA,-1)                       AXIS
   50 GO TO 25                                                         AXIS
   60 FNTIC = NTIC                                                     AXIS
      NT = N*NTIC                                                      AXIS
      TN = NT                                                          AXIS
      CTH = CTN/FNTIC                                                  AXIS
      STH = STN/FNTIC                                                  AXIS
      XB = X + TN*CTH                                                  AXIS
      YB = Y + TN*STH                                                  AXIS
      XDELT = - H(6 - NTIC) * SIG  * STH                               AXIS
      YDELT = H(6 - NTIC) * SIG  * CTH                                 AXIS
      XA = XB + XDELT + XDELT                                          AXIS
C     XA = XB- 0.1 * SIG  * STH                                        AXIS
      YA = YB + YDELT + YDELT                                          AXIS
C     YA = YB+ 0.1 * SIG  * CTH                                        AXIS
      CALL PLOT (XA,YA,3)                                              AXIS
      XA = XA - XDELT                                                  AXIS
      YA = YA - YDELT                                                  AXIS
C     DO 20  I =1,N                                                    AXIS
      DO 20  I =1,N                                                    AXIS
      DO 20 II =1,NTIC                                                 AXIS
      IF(II.LT.NTIC) GO TO 45                                          AXIS
      XX = XDELT                                                       AXIS
      YY = YDELT                                                       AXIS
      GO TO 46                                                         AXIS
   45 XX = 0.                                                          AXIS
      YY = 0.                                                          AXIS
   46 CONTINUE                                                         AXIS
      CALL PLOT (XB,YB,2)                                              AXIS
      XC = XB - CTH                                                    AXIS
      YC = YB - STH                                                    AXIS
      CALL PLOT (XC,YC,2)                                              AXIS
```

```
C       XA = XA - CTH                                                   AXIS
        XA = XA - CTH + XX                                              AXIS
C       YA = YA - STH                                                   AXIS
        YA = YA - STH + YY                                              AXIS
        CALL PLOT (XA,YA,2)                                             AXIS
        XA = XA - XX                                                    AXIS
        YA = YA - YY                                                    AXIS
        XB = XC                                                         AXIS
   20   YB = YC                                                         AXIS
        RETURN                                                          AXIS
        END                                                             AXIS
        SUBROUTINE SCALE (X,S,N,K)                                      SCALE
C                                                                       SCALE
C  WHERE-  X  IS THE NAME OF THE ARRAY OF DATA TO BE SCANNED FOR MAXIMUMSCALE
C             AND MINIMUM VALUES. AN ADJUSTED MINIMUM VALUE WILL BE     SCALE
C             STORED IN X(N*K+1). AN ADJUSTED DX(MAX.-MIN.) WILL BE     SCALE
C             STORED IN X(N*K+K+1).                                     SCALE
C          S  IS THE LENGTH OVER WHICH THIS DATA IS TO BE PLOTTED.      SCALE
C          N  IS THE NUMBER OF DATA POINTS IN THE ARRAY X.              SCALE
C          K  IS THE REPEAT CYCLE OF A MIXED ARRAY.(NORMALLY 1)         SCALE
C                                                                       SCALE
        DIMENSION X(2)                                                  SCALE
        IT = 13                                                         SCALE
        NP = N * K                                                      SCALE
        L = NP + 1                                                      SCALE
        J = NP +K + 1                                                   SCALE
C       XMAX = X(1)                                                     SCALE
C       X(L) = X(1)                                                     SCALE
        DO 2 I=1,NP,K                                                   SCALE
        IF(X(I)) 2,2,3                                                  SCALE
    2   CONTINUE                                                        SCALE
    3   XMAX = X(1)                                                     SCALE
C       X(L)= X(I)                                                      SCALE
        X(L)= 0.0                                                       SCALE
        DO 10 I = 1,NP,K                                                SCALE
        IF(X(I)) 22,10,23                                               SCALE
   22   ITEST = 2.5 - X(I)                                              SCALE
        IF(ITEST) 10,24,10                                              SCALE
   24   X(L) = 0.0                                                      SCALE
        IT = 0                                                          SCALE
        GO TO 10                                                        SCALE
   23   IF (XMAX-X(I)) 5,6,6                                            SCALE
    5   XMAX = X(I)                                                     SCALE
    6   IF (X(L)-X(I))10,10,7                                           SCALE
    7   X(L) = X(I)                                                     SCALE
   10   CONTINUE                                                        SCALE
        DX = (XMAX - X(L)) /S                                           SCALE
        IF (DX) 31,31,30                                                SCALE
   31   X(J) = 1.0                                                      SCALE
        X(L) = X(L) - 0.5                                               SCALE
        RETURN                                                          SCALE
   30   IDX = ALOG10 (DX)                                               SCALE
        IXMN = X(L) * 10.0 ** (-IDX)                                    SCALE
        IF (X(L))32,33,34                                               SCALE
   32   IXMN = X(L) * 10.0 ** (-IDX) - 0.99                             SCALE
   34   X(L) = IXMN                                                     SCALE
        X(L) = X(L) * 10.0 ** IDX                                       SCALE
   33   DX = ALOG10 ((XMAX-X(L))/S)                                     SCALE
        IDX = DX                                                        SCALE
        XMAX = IDX                                                      SCALE
```

```
      DX=10.0**(LX-XMAX)                                                SCALE
      XMAX = 1.0                                                        SCALE
   41 IF (IX-1.0) 40,20,11                                              SCALE
   40 DX = DX * 10.0                                                    SCAL
      IDX = IDX - 1                                                     SCAL
      GO TO 41                                                          SCALE
   11 XMAX=2.0                                                          SCALE
      IF(DX-2.0) 20,20,12                                               SCALE
   12 XMAX = 4.0                                                        SCALE
      IF (DX-4.0) 20,20,13                                              SCALE
   13 XMAX=5.0                                                          SCALE
      IF(DX-5.0) 20,20,14                                               SCALE
   14 XMAX=8.0                                                          SCALE
      IF (DX-8.0) 20,20,15                                              SCALE
   15 XMAX=10.0                                                         SCALE
   20 X(J) = XMAX * 10.0 ** IDX                                         SCALE
      IF(IT) 49,49,39                                                   SCALE
   39 CONTINUE                                                          SCALE
      IMAX = XMAX + 0.00001                                             SCALE
      IXMN = X(L) * 10.0 ** (-IDX)                                      SCALE
      IMUL = IXMN/IMAX*IMAX                                             SCALE
      X(L) = IMUL                                                       SCALE
      X(L) = X(L) * 10.0 ** IDX                                         SCALE
      GO TO 59                                                          SCALE
   49 CONTINUE                                                          SCALE
      X(L) = 0.0                                                        SCALE
   59 CONTINUE                                                          SCALE
      RETURN                                                            SCALE
      END                                                               SCAL
      SUBROUTINE RET                                                    RET
C                                                                       RET
C MSC ENDOCRINE DATA RETREIVAL PROGRAM                                  RET
C     FIRST LOOK STATISTICS PROGRAM.                                    RET
C                                                                       RET
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP        XREDUC
C     COMMON /INDEX/ INDX(124),INDX1(124),INDX2(248),INDX3(124)         RET
      COMMON/HITBLK/LHIT(150),NDATE(150),ISAV (16),MTVOL(150)           XREDUC
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),   RET
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA           RET
C     COMMON /S1SBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,      RET
      COMMON /S1SBUF/ JBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,      RET
     +                MFUNC,IFILE,IERR                                  RET
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)           RET
      DIMENSION IBUF(14)                                                RET
      EQUIVALENCE(IBUF,T)                                               RET
      EXTERNAL ERROR                                                    RET
C                                                                       RET
C RA1=SAMPLE DIRECTORY  ***  RA2=URINE FILE  ***  RA3=BLOOD FILE        RET
C                       ***  RA4=DAILY FILE  ***                       RET
C                                                                       RET
C                                                                       RET
C PLACE KEYED FILES IN UPDATE MODE EXCLUDING @RA1@ WHICH IS             RET
C             TO BE USED AS A SEQUENTIAL SEARCH FILE                    RET
C             ASSIGNED TO FORTRAN UNIT NO. 1                            RET
C                                                                       RET
      IF(IFFF) 31,201,140                                               RET
C  31 CALL RANDOM(4,@RA2@,125,5,7,100 ,100 ,ICODE,LOC)                  RET
   31 CONTINUE                                                          RET
      CALL FILLIS(FITRA1,3LLFN,3LRA1,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL, 350,3LMNR, 350,3LWSA, T)                       RET
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      CALL STOREF(FITRA1,3LFRL,20)                                      RET
      CALL STOREF(FITRA1,2LFX,  ERROR)                                  RET
      CALL OPENM(FITRA1,3LI-O)                                          RET
      CALL REWND(FITRA1)                                                RET
      CALL FILEIS(FITRA2,3LIFN,3LRA2,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL,1250,3LMNR,1250,3LWSA, T)                       RET
      CALL STOREF(FITRA2,3LFRL,0)                                       RET
      CALL STOREF(FITRA2,2LFX,  ERROR)                                  RET
      CALL OPENM(FITRA2,5LINPUT)                                        RET
      CALL REWND(FITRA2)                                                RET
      CALL FILEIS(FITRA3,3LIFN,3LRA3,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL, 350,3LMNR, 350,3LWSA, T)                       RET
      CALL STOREF(FITRA3,3LFRL,0)                                       RET
      CALL STOREF(FITRA3,2LFX,  ERROR)                                  RET
      CALL OPENM(FITRA3,5LINPUT)                                        RET
      CALL REWND(FITRA3)                                                RET
      CALL FILEIS(FITRA4,3LIFN,3LRA4,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL, 300,3LMNR, 300,3LWSA,T)                        RET
      CALL STOREF(FITRA4,3LFRL,0)                                       RET
      CALL STOREF(FITRA4,2LFX,  ERROR)                                  RET
      CALL OPENM(FITRA4,5LINPUT)                                        RET
      CALL REWND(FITRA4)                                                RET
      IFFF = 0                                                          RET
C                                                                       RET
C INITIALIZE COUNTER FLAGS FOR THE FIRST PASS OF                        RET
C    ANY GIVEN RETREIVAL                                                RET
C                                                                       RET
  201 ISFLG = 1                                                         RET
      IDFLG = 1                                                         RET
      IMFLG = 1                                                         RET
      ITPFLG = 1                                                        RET
      ITTFLG = 1                                                        RET
      IFFF = 1                                                          RET
C                                                                       RET
C READ ONE DATA CARD                                                    RET
C                                                                       RET
    1 READ 200,(ISAV(I),I=1,16)                                         RET
  200 FORMAT(A4,1X,A4,1X,14I5)                                          RET
C                                                                       RET
C DETERMINE CARD TYPE AND COUNTER SEQUENCE                              RET
C                                                                       RET
      IF(ISAV(1).EQ.4HJOB ) GO TO 19                                    RET
      IF(ISAV(1).EQ.4HSTUD.AND.ISFLG.EQ.1) GO TO  5                     RET
      IF(ISAV(1).EQ.4HSTUD.AND.ISFLG.EQ.0) GO TO 45                     RET
      IF(ISAV(1).EQ.4HDATE.AND.IDFLG.EQ.1) GO TO 6                      RET
      IF(ISAV(1).EQ.4HDATE.AND.IDFLG.EQ.0) GO TO 46                     RET
      IF(ISAV(1).EQ.4HMAN .AND.IMFLG.EQ.1) GO TO  7                     RET
      IF(ISAV(1).EQ.4HMAN .AND.IMFLG.EQ.0) GO TO 47                     RET
      IF(ISAV(1).EQ.4HTYPE.AND.ITPFLG.EQ.1) GO TO  8                    RET
      IF(ISAV(1).EQ.4HTYPE.AND.ITPFLG.EQ.0) GO TO 48                    RET
      IF(ISAV(1).EQ.4HTEST.AND.ITTFLG.EQ.1) GO TO  9                    RET
      IF(ISAV(1).EQ.4HTEST.AND.ITTFLG.EQ.0) GO TO 49                    RET
      IF(ISAV(1).EQ.4HEND ) GO TO 40                                    RET
      IF(ISAV(1).EQ.4HEOF ) GO TO 99                                    RET
C                                                                       RET
C BUILD RETREIVAL CRITERIA ARRAYS                                       RET
C                                                                       RET
    5 JSTALL=0                                                          RET
      IS = 0                                                            RET
      ISFLG = 0                                                         RET
```

```
   45 CALL SETCRI(LSTUDY,IS,JSTALL)                                          RET
C     OUTPUT,(LSTUDY(I),I=1,5),IS,JSTALL                                     RET
      GO TO 1                                                               RET
    6 JDTALL=0                                                              RET
      ID = 0                                                                RET
      IDFLG = 0                                                             RET
   46 CALL SETCRI(LDATE,ID,JDTALL)                                          RET
C     OUTPUT,(LDATE(I),I=1,10),ID,JDTALL                                     RET
      GO TO 1                                                               RET
    7 MANALL=0                                                              RET
      IM = 0                                                                RET
      IMFLG = 0                                                             RET
   47 CALL SETCRI(LMAN,IM,MANALL)                                           RET
C     OUTPUT,(LMAN(I),I=1,9),IM,MANALL                                       RET
      GO TO 1                                                               RET
    8 JTPALL=0                                                              RET
      ITP = 0                                                               RET
      ITPFLG = 0                                                            RET
   48 CALL SETCRI(LTYPE,ITP,JTPALL)                                         RET
C     OUTPUT,(LTYPE(I),I=1,10),ITP,JTPALL                                    RET
      GO TO 1                                                               RET
    9 JTTALL=0                                                              RET
      ITT = 0                                                               RET
      ITTFLG = 0                                                            RET
   49 CALL SETCRI(LTEST,ITT,JTTALL)                                         RET
      ITT=1                                                                 RET
C     IF(JTTALL.EQ.1) ITT = 124                                              RET
      LSTART = ISAV(3)                                                      RET
      LSTOP  = ISAV(4)                                                      RET
      IF(LSTOP.EQ.0) LSTOP=LSTART                                           DEC
      GO TO 1                                                               RET
   19 DO 119 I=1,16                                                         RET
  119 JOB(I)=ISAV(I)                                                        RET
      IF(IOPLOT.EQ.0.AND.JOB(3).GE.1000) CALL PLOTS(14HNASA ENDOCRINE,       RET
     1  14)                                                                  RET
      IF(JOB(3).GE.1000) IOPLOT = 999                                       RET
      IF(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH.OR.JOB(2).EQ.4HVOLU)GOTO1     DEC10
      JOB(2)=4HDATE                                                         DEC10
      PRINT 601                                                             DEC10
  601 FORMAT(* UNKNOWN JOB PARAMETER DATE ASSUMED*)                         DEC10
      GO TO 1                                                               RET
C                                                                           RET
C BUILD HIT ARRAY OF SAMPLE NUMBERS                                         RET
C                                                                           RET
   40 CONTINUE                                                              RET
      LT = LSTART - 1                                                       RET
C                                                                           RET
C     DO 4321 LT=LSTART,LSTOP                                               RET
  140 IF(IFFF - 2) 145,240,145                                              RET
  145 LT = LT + 1                                                           RET
C                                                                           RET
      IF(LT - LSTOP) 150,150,4321                                           RET
  150 LM = 0                                                                RET
      IFFF = 2                                                              RET
C     IF(KOT.EQ.0) CALL SSWTCH(1,KKT)                                        RET
C     IF(KKT.EQ.1) GO TO 999                                                 RET
      LTEST(1) = LT                                                         RET
C     DO 4321 LM=1,3                                                        RET
  240 LM = LM + 1                                                           RET
      IF(LM - 3) 250,250,245                                                RET
```

```
  245 IFFF = 1                                                          RET
      GO TO 140                                                         RET
  250 LMAN(1) = LM                                                      RET
      IH = 1                                                            RET
C     REWIND 1                                                          RET
      CALL REWND(FITRA1)                                                RET
C                                                                       RET
C                                                                       RET
C THE KEYED (RA1) SAMPLE DIRECTORY FILE IS TREATED AS A SEQUENTIAL      RET
C     FILE DURING THE SEARCH FOR SAMPLES WHICH SATISFY                  RET
C                    THE INPUT CRITERIA                                 RET
C                                                                       RET
C   10 BUFFER IN(1,1)(IBUF(1),IBUF(14))                                 RET
   10 CALL GETN(FITRA1)                                                 RET
      IF(IFETCH(FITRA1,2LFP).EQ.1008) GO TO 23                          RET
    2 IF(JSTALL.EQ.1) GO TO 12                                          RET
      DO 11 I=1,IS                                                      RET
      IF(IBUF(2).EQ.LSTUDY(I)) GO TO 12                                 RET
   11 CONTINUE                                                          RET
      GO TO 10                                                          RET
   12 IF(JOTALL.EQ.1.AND.JOB(2).EQ.4HVOLD) GO TO 44                     RET
C                                                                       RET
C CHECK FOR SIMULTANEOUS @ALL@ AND DATE @SAVE@ OPTION                   RET
C                                                                       RET
      IF(JOTALL.EQ.1.AND.(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH))GO TO 41RET
      DO 13 I=1,ID                                                      RET
      IF(IBUF(3).EQ.LDATE(I)) GO TO 14                                  RET
   13 CONTINUE                                                          RET
      GO TO 10                                                          RET
   41 MDSAV = IBUF(3)                                                   RET
      GO TO 44                                                          RET
C                                                                       RET
C CHECK FOR DATE SAVE OPTION                                            RET
C                                                                       RET
   14 IF(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH) MDSAV=LDATE(I)           RET
   44  IF(MANALL.EQ.1) GO TO 16                                         RET
      DO 15 I=1,IM                                                      RET
      IF(IBUF(4).EQ.LMAN(I)) GO TO 16                                   RET
   15 CONTINUE                                                          RET
      GO TO 10                                                          RET
   16 IF(JTPALL.EQ.1) GO TO 18                                          RET
      DO 17 I=1,ITP                                                     RET
      IF(IBUF(5).EQ.LTYPE(I)) GO TO 18                                  RET
   17 CONTINUE                                                          RET
      GO TO 10                                                          RET
C                                                                       RET
C CHECK FOR MASTER SAMPLE NUMBER                                        RET
C                                                                       RET
   18 IF(IBUF(14).EQ.0) GO TO 21                                        RET
      MIH = IH-1                                                        RET
      DO 20 I=1,MIH                                                     RET
      IF(LHIT(I).EQ.IBUF(14)) GO TO 10                                  RET
   20 CONTINUE                                                          RET
      LHIT(IH)=IBUF(14)                                                 RET
C     PRINT 220,(LHIT(I),I=1,10)                                        RET
C 220 FORMAT(5X,5HLHIT2,5X,10I10)                                       RET
      GO TO 22                                                          RET
   21 LHIT(IH)=IBUF(1)                                                  RET
C     PRINT 221,(LHIT(I),I=1,10)                                        RET
C 221 FORMAT(5X,5HLHIT1,5X,10I10)                                       RET
```

```
C                                                                       RET
C  IF DATE SAVE OPTION WAS SPECIFIED FILL DATE ARRAY CORRESPONDING      RET
C     TO SAMPLE NUMBER HIT ARRAY                                        RET
C                                                                       RET
   22 IF(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH) MDATE(IH)=MDSAV          RET
C                                                                       RET
C  IF TOTAL VOLUME SAVE OPTION WAS SPECIFIED FILL VOLUME ARRAY          RET
C        CORRESPONDING TO SAMPLE NUMBER HIT ARRAY                       RET
C                                                                       RET
      IF(JOB(2).EQ.4HVOLU.OR.JOB(2).EQ.4HBOTH)MTVOL(IH)=IBUF(10)        RET
       IH=IH+1                                                          RET
      IF(IH.LE.151)  GO TO 10                                           XREDUC
   23 NSMP = IH - 1                                                     RET
      IF(NSMP.LE.150)  GO TO 35                                         XREDUC
      PRINT 30                                                          RET
   30 FORMAT(1H1,17HHIT FILE OVERFLOW    )                              RET
      STOP 30                                                           RET
C                                                                       RET
   35 CALL RETRVE                                                       RET
      PRINT 300, LSTUDY,LMAN,JOB,LTYPE,NSMP,NDATA                       RET
  300 FORMAT(1H0,6HLSTUDY,10I10,//,2X,4HLMAN,9I10,//,                   DEC10
     .   1X, 4HJOB ,A4,5X,A4,5X,14I6,//,6H LTYPE,20I5,//,              RET
     .   1X, 7HNSMP = ,I5,10X,8HNDATA = ,I5)                           DEC10
      PRINT 301,(LDATE(I),I=1,ID)                                       RET
  301 FORMAT(6H LDATE /,(10I10))                                        RET
      PRINT 302,(LTEST(I),I=1,ITT)                                      RET
  302 FORMAT(6H LTEST  ,(10I10))                                        RET
      PRINT 306,(MTVOL(I),I=1,NSMP)                                     RET
  306 FORMAT(1H0,5HMTVOL,/,(10I10))                                     DEC10
      PRINT 303,(MDATE(I),I=1,NSMP)                                     RET
  303 FORMAT(1H0,5HMDATE,/,(10I10))                                     DEC10
      PRINT 304,(LHIT(I),I=1,NSMP)                                      RET
  304 FORMAT(/ ,6H LHIT ,/,(10I10))                                     DEC10
      PRINT 305,(XDATA(I),I=1,NDATA)                                    RET
  305 FORMAT(1H0,5HXDATA,/,(8F15.6))                                    DEC10
C     PRINT 300, LSTUDY,LMAN,JOB,LTYPE,NSMP,NDATA                       RET
C 300 FORMAT(1H1,6HLSTUDY,10I10,//,2X,4HLMAN,9I10,//,                   RET
C    .       4HJOB ,A4,5X,A4,5X,14I6,//,5HLTYPE,20I5,//,                RET
C    .       7HNSMP = ,I5,10X,8HNDATA = ,I5,// )                        RET
C     PRINT 301,(LDATE(I),I=1,ID)                                       RET
C 301 FORMAT(5HLDATE /,(10I10))                                         RET
C     PRINT 302,(LTEST(I),I=1,ITT)                                      RET
C 302 FORMAT(5HLTEST  ,(10I10))                                         RET
C     PRINT 306,(MTVOL(I),I=1,NSMP)                                     RET
C 306 FORMAT(1H1,5HMTVOL,/,(10I10))                                     RET
C     PRINT 303,(MDATE(I),I=1,NSMP)                                     RET
C 303 FORMAT(1H1,5HMDATE,/,(10I10))                                     RET
C     PRINT 304,(LHIT(I),I=1,NSMP)                                      RET
C 304 FORMAT(//,5HLHIT ,/,(10I10))                                      RET
C     PRINT 305,(XDATA(I),I=1,NDATA)                                    RET
C 305 FORMAT(1H1,5HXDATA,/,(8F15.6))                                    RET
C                                                                       RET
C                                                                       RET
C     ------------------------------------------------------------------RET
      RETURN                                                            RET
C     CALL STAT                                                         RET
 4321 CONTINUE                                                          RET
      IFFF = 0                                                          RET
C                                                                       RET
C     ------------------------------------------------------------------RET
```

```
C                                                                       RET
C     REWIND 1                                                          RET
      CALL REWND(FITRA1)                                                RET
      GO TO 201                                                         RET
C                                                                       RET
  888 PRINT 8800                                                        RET
 8800 FORMAT(15H FILE 1 REWOUND)                                        RET
      RETURN                                                            RET
   99 CONTINUE                                                          RET
      CALL CLOSEM(FITRA1)                                               RET
      CALL CLOSEM(FITRA2)                                               RET
      CALL CLOSEM(FITRA3)                                               RET
      CALL CLOSEM(FITRA4)                                               RET
      IF(IOPLOT) 199,199,999                                            RET
  999 CALL STOPPLT                                                      RET
      PRINT 900                                                         RET
  900 FORMAT(19HONORMAL END OF JOB.)                                    RET
      STOP 2000                                                         RET
  199 STOP 1                                                            RET
      END                                                               RET
      SUBROUTINE SETCRI(LBUF,JNDX,LALL)                                 SETCRI
C  THIS SUBROUTINE STORES INFORMATION FROM ONE DATA CARD INTO THE       SETCRI
C                          CORRESPONDING CRITERIA ARRAY                 SETCRI
C                                                                       SETCRI
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)          SETCRI
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTYPE(150)           XREDUC
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),   SETCRI
     +                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA           SETCRI
      COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,      SETCRI
     +                MFUNC,IFILE,IERR                                  SETCRI
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)           SETCRI
      DIMENSION LBUF(1)                                                 SETCRI
      IF(ISAV(2).EQ.4HALL .OR.ISAV(2).EQ.4H    ) GO TO 1                SETCRI
      JNDX = JNDX + 1                                                   SETCRI
      LBUF(JNDX)=ISAV(3)                                                SETCRI
      IF(ISAV(2).EQ.4HEACH) GO TO 3                                     SETCRI
      INXT=ISAV(3)+1                                                    SETCRI
    2 IF(INXT.GT.ISAV(4)) RETURN                                        SETCRI
      JNDX = JNDX + 1                                                   SETCRI
      LBUF(JNDX)=INXT                                                   SETCRI
      INXT=INXT+1                                                       SETCRI
      GO TO 2                                                           SETCRI
    3 DO 5 I=4,16                                                       SETCRI
      IF(ISAV(I).EQ.0) RETURN                                           SETCRI
      JNDX = JNDX + 1                                                   SETCRI
      LBUF(JNDX)=ISAV(I)                                                SETCRI
    5 CONTINUE                                                          SETCRI
      RETURN                                                            SETCRI
    1 LALL=1                                                            SETCRI
      RETURN                                                            SETCRI
      END                                                               SETCRI
      SUBROUTINE RETRVE                                                 RETRVE
C                                                                       RETRVE
C THIS SUBROUTINE RETREVES THOSE DATA VALUES FROM THE BLOOD OR          RETRVE
C      URINE FILES WHICH SATISFY THE INPUT CRITERIA                     RETRVE
C      THE DATA VALUES ARE STORED IN ARRAY XDATA                        RETRVE
C                                                                       RETRVE
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP        XREDUC
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)          RETRVE
      COMMON/INDEX/INDX(136),INDX3(136)                                 DEC10
```

```
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)            XREDUC
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),    RETRVE
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA            RETRV
      COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEE,KYE,       RETR
     +                MFUNC,IFILE,IERR                                   RETRV
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)            RETRVE
      DIMENSION XTEMP(125)                                               RETRVE
      INTEGER KEY(2)                                                     RETRVE
      EQUIVALENCE(T,XTEMP)                                               RETRVE
      EQUIVALENCE (KEE,KEY(2))                                           RETRVE
      DATA(INDX(I),I=1,136)/400,401,402,403,404,405,503,406,407,408,409,DEC10
     .  410,                                                             DEC10
     1  411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,426,DEC10
     2  427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,DEC10
     3  443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,DEC10
     4  459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,DEC10
     5  475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,490,DEC10
     6  491,492,493,494,495,496,497,498,499,500,501,502,300,301,302,303,DEC10
     7  304,305,306,307,321,308,309,310,311,312,313,314,315,316,317,318,DEC10
     .  319,                                                             DEC10
     8  320,514,515,516,517,518,519,520,521,522,523/                    DEC10
      DATA (INDX3(I),I=1,136)/2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,   DEC10
     1  18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,  DEC10
     2  39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,  DEC10
     3  60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,  DEC10
     4  81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,    DEC10
     5101,102,103,104,105,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,DEC10
     6  20,21,22,23,116,117,118,119,120,121,122,123,124,125/             DEC10
      DATA ICDT/445B/                                                    RETRVE
      M=1                                                                RETR
      KSAV=0                                                             RETRV
      IERR=0                                                             RETRVE
      DO 27 K=1,NSMP                                                     RETRVE
      DO 27 I=1,ITT                                                      RETRVE
C                                                                        RETRVE
C  IDENTIFY TEST NUMBER IN INDEX                                         RETRVE
C                                                                        RETRVE
      DO 24 J=1,136                                                      DEC10
      IF(LTEST(I).EQ.INDX(J)) GO TO 25                                   DEC10
   24 CONTINUE                                                           DEC10
      STOP @INVALID TEST NUMBER@                                         DEC10
   25 IJ=J                                                               DEC10
      LREC=INDX3(IJ)                                                     DEC10
      IF(LTEST(I) -400) 41,31,31                                         RETRVE
   31 INDX1 = 1RU                                                        RETRVE
      GO TO 50                                                           RETRVE
   41 INDX1 = 1RB                                                        RETRVE
C                                                                        RETRVE
C  PICK UP KEY FROM HIT ARRAY OF SAMPLE NUMBERS                          RETRVE
C                                                                        RETRVE
   50 KEY(2)=LHIT(K)                                                     RETRVE
C                                                                        RETRVE
C DETERMINE IF DATA NEEDED IS FROM CURRENT RECORD                        RETRVE
C                                                                        RETRVE
      IF(KEY(2).EQ.KSAV) GO TO 26                                        RETRVE
C                                                                        RETR
C  CHECK FOR URINE OR BLOOD FILE                                         RETRVE
C                                                                        RETRVE
C  29 IF(INDX1(IJ).EQ.1RU) GO TO 28                                      RETRVE
   29 IF(INDX1.EQ.1RU) GO TO 28                                          RETRVE
```

```
C       CALL RANDOM(1,@RA3@,XTEMP,32,KEY,100 ,100 ,ICODE,LOC)                RETRVE
        CALL GET(FITRA3)                                                     RETRVE
        LOC = 3                                                              RETRVE
        IFILE = 3                                                            RETRVE
        IF(IERR.NE.0) GO TO 100                                              RETRVE
        GO TO 26                                                             RETRVE
C   28 CALL RANDOM(1,@RA2@,XTEMP,125,KEY,100 ,100 ,ICODE,LOC)                RETRVE
    28 CALL GET(FITRA2)                                                      RETRVE
        LOC = 2                                                              RETRVE
        IFILE = 2                                                            RETRVE
        IF(IERR.NE.0) GO TO 100                                              RETRVE
C                                                                            RETRVE
C                                                                            RETRVE
C FETCH WORD NUMBER OF DATA RECORD FROM INDEX, STORE DATA                    RETRVE
C    VALUE IN CURRENT XDATA LOCATION AND INCREMENT                           RETRVE
C            XDATA LOCATION COUNTER M                                        RETRVE
C                                                                            RETRVE
C                                                                            RETRVE
C   26 LREC=INDX3(IJ)                                                        RETRVE
    26 XDATA(M)=XTEMP(LREC)                                                  RETRVE
    30 M=M+1                                                                 RETRVE
        IF(M.GT.150)  GO TO 110                                              XREDUC
        KSAV=KEY(2)                                                          RETRVE
    27 CONTINUE                                                              RETRVE
C                                                                            RETRVE
C                                                                            RETRVE
C NDATA = NUMBER OF DATA VALUES RETREIVED                                    RETRVE
C                                                                            RETRVE
C                                                                            RETRVE
        NDATA=M-1                                                            RETRVE
        RETURN                                                               RETRVE
  100 IF(IERR .EQ.ICDT) GO TO 105                                            RETRVE
        PRINT 101,IERR ,KEY(2)                                               RETRVE
  101 FORMAT(1H ,19HRANDOM ERROR-RETRVE,08,5X,I4)                            RETRVE
        STOP 100                                                             RETRVE
  105 PRINT 104,KEY(2)                                                       RETRVE
  104 FORMAT(1H ,27HSAMPLE NO. NOT FOUND-RETRVE,5X,I4)                       RETRVE
        XDATA(M)=-1.                                                         RETRVE
        IERR=0                                                               RETRVE
        GO TO 30                                                             RETRVE
  110 PRINT 111                                                              RETRVE
  111 FORMAT(1H1,43HDATA BUFFER FULL BEFORE ALL DATA RETREIVED    )          RETRVE
        STOP 110                                                             RETRVE
        END                                                                  RETRVE
        SUBROUTINE STAT                                                      STAT
        COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP           XREDUC
        COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)              XREDUC
        COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),      STAT
     .                  LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA              STAT
        COMMON/RG/ DATESC(3,5),IVV(4), SSMEAN(3), SSIGMA(3), NNN(3),SSE(3)   STAT
        COMMON Y(3,3,80),N(3,3),A(4,4),B(4,4),AP(3),Q(3),W(3),V(3),WW(3),    ANOV1
       1VV(3),NSUMI(3),SUMYI(3),NSUMJ(3),SUMYJ(3),SUMYIJ(3,3),TAUHAT(3),     ANOV1
       2    BETHAT(3),THAT(3),BHAT(3)                                        ANOV1
        COMMON/AOV2NE/IDEN(8),NAMET,NAMEB,IWT,SUMSQX,TAU,MNSQT,FRATIO,       ANOV1
       1 IERR,ERROR,MNSQE,SST,MNSST,FSST,INT,TIN,TINT,FINT,ITHT,WITHT,       ANOV1
       2 WIT,NSUM,SUMSQY,SUMSQT,BETA,MNSQB,FRATB,ERRORB,MNSQBE,SSB,MNSSB,    ANOV1
       3 FSSB,BIN,BINT,FINB,PFRAT,PFSST,PFINT,PFRATB,PFSSB,PFINB             ANOV1
        REAL KSTUDY, KTYPE, KMAN, KTEST, NITS                                STAT
        REAL MNSSB,MNSST,MNSQB,MNSQBE,MNSQT,MNSQE                            ANOV1
        DIMENSION DEN(8)                                                     ANOV1
```

```
      DIMENSION KOE(3)                                                  ANOV1
      EXTERNAL TWOWT,TWOAVT,TWOAVB                                      ANOV1
      DIMENSION ISIG(2)                                                 STAT
      DIMENSION X(1)                                                    STAT
      DIMENSION LTEXT(120), MTEXT(120), XOUT(120)                       STAT
      DIMENSION NDF(3), S(3), P(3), TM(3), WTV(3), XMS(3), IPP(3),IP2(3)STAT
      DIMENSION ID1(120), ID2(120), LK(3)                               STAT
      EQUIVALENCE (IDEN,DEN)                                            ANOV
      EQUIVALENCE(X,XDATA)                                              STAT
      DATA ID1/120*1/, LK/3*0/                                          STAT
      DATA IPP/4HPRE , 4HIN  , 4HPOST/                                  STAT
      DATA IP2/8HPRE-IN  , 8HIN-POST , 8HPRE-POST/                      STAT
C                                                                       STAT
C     FILL OUTPUT ARRAY                                                 STAT
C                                                                       STAT
      DATA ISTAR/1H*/, IBLANK/1H /, ISAVE/4HSAVE/                       STAT
      DATA IPLUS/1H+/, IMINUS/1H-/, IPPL/2H++/, IMMI/2H--/              STAT
      DATA IDATE/4HDATE/, IVOLU/4HVOLU/, IBOTH/4HBOTH/                  STAT
      NOBS = NDATA                                                      STAT
    1 IF(NOBS - 1) 999,999,2                                            STAT
    2 LOOP = ITT                                                        STAT
      IDATEF = 0                                                        STAT
      IVOLUF = 0                                                        STAT
      IF(JOB(2).EQ.IDATE) IDATEF = 3                                    STAT
      IF(JOB(2).EQ.IVOLU) IVOLUF = 3                                    STAT
      IF(JOB(2).NE.IBOTH) GO TO 3                                       STAT
      IDATEF = 3                                                        STAT
      IVOLUF = 3                                                        STAT
    3 KKK = NOBS/LOOP                                                   STAT
      CALL TEXT(LSTUDY(1), LTYPE(1), LMAN(1), LTEST(1),                 STAT
     X KSTUDY, KTYPE, KMAN, KTEST, NITS)                                STAT
      CALL DATE(IWORD)                                                  STAT
      PRINT 1000, JOB(3)                                                STAT
 1000 FORMAT(1H1,20X,42HNASA MSC ENDOCRINE DATA RETREIVAL PROGRAM.,I7/) STAT
      PRINT 500, LSTUDY(1), KSTUDY,IWORD                                STAT
  500 FORMAT(7HOSTUDY.,I2,2X, A8,20X,A10)                               STAT
      PRINT 600, LTEST(1), KTEST,LMAN(1), KMAN                          STAT
  600 FORMAT(6HOTEST.,I3,2X A8 ,20X,4HMAN.,I4,2X, A8)                   STAT
      PRINT 700,(LTYPE(I),I=1,ITP)                                      DEC10
  700 FORMAT(6HOTYPE.,20I3)                                             DEC10
      PRINT 800, NITS                                                   STAT
  800 FORMAT(15HOJULIAN DATE     , A8)                                  STAT
C     DO 150 L=1,LOOP                                                   STAT
      L = 1                                                             STAT
      LOOP = 1                                                          STAT
      DO 210 II=LOOP,NOBS,LOOP                                          STAT
      I = II + L - LOOP                                                 STAT
      IPLACE = II/LOOP                                                  STAT
  210 XOUT(IPLACE) = XDATA(I)                                           STAT
C                                                                       STAT
C     SORT BY DATE                                                      STAT
C                                                                       STAT
      NOSLTS = NOBS/LOOP                                                STAT
      NM1 = NOSLTS - 1                                                  STAT
      DO 230 I=1,NM1                                                    STAT
      IMAX = 9999                                                       STAT
      ISUB = I                                                          STAT
      DO 220 J=I,NOSLTS                                                 STAT
      IF(IMAX-MDATE(J)) 220,220,211                                     STAT
  211 IMAX = MDATE(J)                                                   STAT
```

```
      ISUB = J                                                          STAT
  220 CONTINUE                                                          STAT
      ITEMP = MDATE(ISUB)                                               STAT
      TEMP = XOUT(ISUB)                                                 STAT
      MDATE(ISUB) = MDATE(I)                                            STAT
      XOUT(ISUB) = XOUT(I)                                              STAT
      MDATE(I) = ITEMP                                                  STAT
      XOUT(I) = TEMP                                                    STAT
  230 CONTINUE                                                          STAT
      IG = LSTUDY(1)                                                    STAT
      IVV(1)=IVV(2)=IVV(3)=0                                            APR24
      IVV(4) = NOSLTS                                                   STAT
      DO 240 I=1,NOSLTS                                                 STAT
      IF(MDATE(I).LE.DATESC(2,IG)) IVV(2) = I                           STAT
      IF(MDATE(I).LE.DATESC(3,IG)) IVV(3) = I                           STAT
  240 XDATA(I) = XOUT(I)                                                STAT
      DO 133 I=1,3                                                      STAT
      NFIRST = IVV(I) + 1                                               STAT
      NLAST = IVV(I + 1)                                                STAT
      CALL MESIG(LTEXT,NFIRST,NLAST,SMEAN,SIGMA,NN)                     STAT
      SSMEAN(I) = SMEAN                                                 STAT
      SSIGMA(I) = SIGMA                                                 STAT
      NNN(I) = NN                                                       STAT
  133 CONTINUE                                                          STAT
      NFIRST = 1                                                        STAT
      NLAST = NOSLTS                                                    STAT
      CALL MESIG(LTEXT,NFIRST,NLAST,SMEAN,SIGMA,NN)                     STAT
C                                                                       STAT
C     IF THERE ARE 2 OR MORE PRE-FLIGHT OBSERVATIONS THE EXTREME VALUES STAT
C     ARE FLAGED ON THE BASIS OF PRE-FLIGHT MEAN AND STD. DEV.          STAT
C     OTHERWISE THE TOTAL MEAN AND STD. DEV. ARE USED.                  STAT
C                                                                       STAT
      IF(NNN(1) - 2) 134,136,136                                        STAT
  134 SSG = SIGMA                                                       STAT
      SSM = SMEAN                                                       STAT
      GO TO 137                                                         STAT
  136 SSG = SSIGMA(1)                                                   STAT
      SSM = SSMEAN(1)                                                   STAT
  137 T01 = 2.3263*SSG                                                  STAT
      T05 = 1.6449*SSG                                                  STAT
      DO 166 II=1,3                                                     STAT
      IFIRST = IVV(II)+ 1                                               STAT
      ILAST = IVV(II + 1)                                               STAT
      IF(IFIRST-ILAST)162,163,166                                       APR24
  163 PRINT 401                                                         APR24
  401 FORMAT(*  SINGLE VALUE NOT ANALYZED*)                             APR24
  162 DO 165 I=IFIRST,ILAST                                             STAT
      MTEXT(I) = IBLANK                                                 STAT
      IF(X(I)) 164,165,153                                              STAT
  153 DELTA = X(I) - SSM                                                STAT
      IF(DELTA) 155,154,157                                             STAT
  154 IF(X(I).NE.2.0) GO TO 164                                         STAT
      DELTA = SSM                                                       STAT
      GO TO 156                                                         STAT
  155 DELTA = -DELTA                                                    STAT
  156 IF(DELTA.GE.T05) MTEXT(I) = IMINUS                                STAT
      IF(DELTA.GE.T01) MTEXT(I) = IMMI                                  STAT
      GO TO 164                                                         STAT
  157 IF(DELTA.GE.T05) MTEXT(I) = IPLUS                                 STAT
      IF(DELTA.GE.T01) MTEXT(I) = IPPL                                  STAT
```

```
  164 PRINT 400,MDATE(I), X(I), LTEXT(I), MTEXT(1)                    STAT
  165 CONTINUE                                                        STAT
      PRINT 900,SSMEAN(II), SSIGMA(II), NNN(II)                       STAT
  166 CONTINUE                                                        STAT
      IF(JOB(3) - 1000) 150,145,145                                   STAT
  145 IF(NN) 150,150,146                                              STAT
  146 CALL STPLOT(LOOP,NOBS,NSLOT,IWORD,KSTUDY,KTYPE,KMAN,KTEST,SMEAN, STAT
     X SIGMA)                                                         STAT
  150 CONTINUE                                                        STAT
C                                                                     STAT
C     ICASE = 0     NO TESTS                                        STAT
C     ICASE = 1     PREFLIGHT ONLY                                  STAT
C     ICASE = 2     PRE AND INFLIGHT ONLY                           STAT
C     ICASE = 3     ALL                                             STAT
C     ICASE = 4     PRE AND POSTFLIGHT ONLY                         STAT
C                                                                     STAT
      IF(NNN(1)-1) 170,170,171                                        DEC1
  170 NT = 0                                                          STAT
      ICASE = 0                                                       STAT
      GO TO 999                                                       DEC1
  171 SSE(1) = SSIGMA(1)/SQRT(FLOAT(NNN(1)))                          STAT
      PRINT 3000,NNN(1), SSMEAN(1), SSIGMA(1), SSE(1)                 STAT
 3000 FORMAT(20(4H----)/23H PARAMETRIC STATISTICS.//8HOSUMMARY/       STAT
     X39HOSAMPLE  N     MEAN          SD          SE/4HOPRE, I6, 3F11.3)STAT
      IF(NNN(2)-1) 172,172,176                                        DEC1
  172 IF(NNN(3)-1) 173,173,174                                        DEC1
  173 NT = 1                                                          STAT
      ICASE = 1                                                       STAT
      GO TO 999                                                       DEC1
  174 SSE(3) = SSIGMA(3)/SQRT(FLOAT(NNN(3)))                          STAT
      PRINT3010,NNN(3), SSMEAN(3), SSIGMA(3), SSE(3)                  STAT
      NT = 2                                                          STAT
      ICASE = 4                                                       STAT
      NN=NN-NNN(2)                                                    FIX2
C                                                                     STAT
      GO TO 180                                                       DEC1
C     CHANGE THIS LATER                                               STAT
C                                                                     STAT
  176 SSE(2) = SSIGMA(2)/SQRT(FLOAT(NNN(2)))                          STAT
      PRINT3020,NNN(2), SSMEAN(2), SSIGMA(2), SSE(2)                  STAT
 3020 FORMAT(4H0IN ,I6,3F11.3)                                        STAT
      IF(NNN(3)-1) 177,177,178                                        DEC1
  177 NT = 2                                                          STAT
      ICASE = 2                                                       STAT
      NN=NN-NNN(3)                                                    FIX2
      GO TO 180                                                       STAT
  178 SSE(3) = SSIGMA(3)/SQRT(FLOAT(NNN(3)))                          STAT
      PRINT3010,NNN(3), SSMEAN(3), SSIGMA(3), SSE(3)                  STAT
 3010 FORMAT(5H0POST,I5,3F11.3)                                       STAT
      NT = 3                                                          STAT
      ICASE = 3                                                       STAT
  180 CONTINUE                                                        STAT
      SE = SIGMA/SQRT(FLOAT(NN))                                      STAT
      PRINT 3030, NN, SMEAN, SIGMA, SE                                STAT
 3030 FORMAT(4H ---,10(4H----)/4H SUM,I6,3F11.3)                      STAT
C                                                                     STAT
C     GET RID OF NEGATIVE VALUES                                      STAT
C        I  RANGES OVER ALL POINTS.                                   STAT
C        J  RANGES OVER VALID POINTS.                                 STAT
C                                                                     STAT
```

```
      J = 1                                                              STAT
      DO 330 I=1,NOSLTS                                                  STAT
      IF(X(I)) 310,330,320                                               STAT
  310 IF(X(I) + 2.0) 330,315,330                                         STAT
  315 X(J) = 0.0                                                         STAT
      J = J + 1                                                          STAT
      GO TO 330                                                          STAT
  320 X(J) = X(I)                                                        STAT
      J = J + 1                                                          STAT
  330 CONTINUE                                                           STAT
      IF(ICASE.EQ.4) 710,711                                             FIX
  710 IF(NNN(2).EQ.1) GO TO 712                                          FIX
      GO TO 713                                                          FIX
  712 NNN1=NNN(1)+1                                                      FIX
      NSLTM1=J-1                                                         FIX
      DO 179 I=NNN1,NSLTM1                                               FIX
  179 X(I)=X(I+1)                                                        FIX
  713 NNN(2)=NNN(3)                                                      FIX
      NNN(3)=0                                                           FIX
  711 CONTINUE                                                           FIX
      ITOT=JTOT=3                                                        ANOV1
      IF(ICASE.EQ.2.OR.ICASE.EQ.4) ITOT=2                                ANOV1
      IB=LMAN(1)                                                         ANOV1
      K1=0                                                               ANOV1
      DO 699 IT=1,ITOT                                                   ANOV1
      K=N(IT,IB)=NNN(IT)                                                 ANOV1
      DO 702 I=1,K                                                       ANOV1
  702 Y(IT,IB,I)=X(I+K1)                                                 ANOV1
      K1=K+K1                                                            ANOV1
  699 CONTINUE                                                           ANOV1
      CALL ACRDAN(X, NT, NNN, TM, WTV, S, GM, NDF, IER)                  STAT
      N1 = NDF(1)                                                        STAT
      N2 = NDF(2)                                                        STAT
      S1 = S(1)                                                          STAT
      S2 = S(2)                                                          STAT
      CALL FFOUT(S1,S2,N1,N2,FVAL,PRFVAL)                                STAT
      DO 193 IMS=1,3                                                     STAT
  193 XMS(IMS) = S(IMS)/NDF(IMS)                                         STAT
      PRINT 3090                                                         STAT
 3090 FORMAT(1X)                                                         STAT
      PRINT 3100                                                         STAT
 3100 FORMAT(28H0ANALYSIS OF VARIANCE TABLE./35H0        DF       SS      STAT
     X  MS      F./35H         ----     ----       ----    ---)         STAT
      PRINT 3200, NDF(1), S(1), XMS(1), FVAL                             STAT
 3200 FORMAT(6H TREAT, I4, F10.1, F10.1, F8.3)                           STAT
      PRINT 3300, NDF(2), S(2), XMS(2)                                   STAT
 3300 FORMAT(6H0ERROR, I4, F10.1, F10.1, 6H  --- /4H ---, 9(4H----))     STAT
      PRINT 3400, NDF(3), S(3), XMS(3)                                   STAT
 3400 FORMAT(6H TOTAL, I4, F10.1, F10.1, 6H  --- )                       STAT
      PRINT 3800, FVAL, PRFVAL                                           STAT
 3800 FORMAT(1H0,F7.3, 23H IS SIGNIFICANT AT THE  ,F5.1, 14H PERCENT LEVSTAT
     1EL)                                                                STAT
      IF(NT-2) 999,850,824                                               DEC10
  824 IF(PRFVAL - 5.0) 825,825,850                                       STAT
  825 N1 = 1                                                             STAT
      PRINT 3500                                                         STAT
 3500 FORMAT(14H0CONTRAST    F )                                         STAT
      P(1) = 1                                                           STAT
      P(2) = -1                                                          STAT
      P(3)=0                                                             FIX
```

```
      CALL ACTRST(TM,  NNN, 3, 1, P, 1, Q, SQ)                          STAT
      CALL FFOUT(SQ,S2,N1,N2,FVAL,PRFVAL)                               STAT
      PRINT 3600,IP2(1),FVAL,PRFVAL                                     FIX
 3600 FORMAT(1H0,A8,F8.3,23H IS SIGNIFICANT AT THE   ,F5.1,             FIX
     X15H PERCENT LEVEL.)                                               STAT
      P(1) = 0                                                          STAT
      P(2) = 1                                                          STAT
      P(3) = -1                                                         STAT
      CALL ACTRST(TM,  NNN, 3, 1, P, 1, Q, SQ)                          STAT
      CALL FFOUT(SQ,S2,N1,N2,FVAL,PRFVAL)                               STAT
      PRINT 3600,IP2(2),FVAL,PRFVAL                                     FIX
      P(1) = 1                                                          STAT
      P(2) = 0                                                          STAT
      CALL ACTRST(TM,  NNN, 3, 1, P, 1, Q, SQ)                          STAT
      CALL FFOUT(SQ,S2,N1,N2,FVAL,PRFVAL)                               STAT
      PRINT 3600,IP2(3),FVAL,PRFVAL                                     FIX
  850 CONTINUE                                                          STAT
  851 NP = 0                                                            STAT
      DO 855 I=1,NT                                                     STAT
      NCOC = NNN(I)                                                     STAT
      DO 855 J=1,NCOC                                                   STAT
      NP = NP + 1                                                       STAT
  855 ID2(NP) = I                                                       STAT
      IF(ICASE.EQ.4) GO TO 860                                          DEC1
      PRINT 4000                                                        STAT
 4000 FORMAT(1H0/20(4H----)/27H NON-PARAMETRIC STATISTICS.)             STAT
C     CALL  KRUSWAL(X,ID1,ID2,NP,0,0,NT,NNN,LK)                        STAT
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,NT,NNN,LK,LK)                       FIX2
      PRINT 1100,IP2(1)                                                 STAT
 1100 FORMAT(1H0/10HOCONTRAST , A8)                                     STAT
      NP = NNN(1) + NNN(2)                                              STAT
C     CALL  KRUSWAL(X,ID1,ID2,NP,0,0, 2,NNN,LK)                        STAT
      IPI=2                                                             FIX2
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,2,NNN,LK,IPI)                       FIX2
      IF(ICASE.EQ.2) GO TO 998                                          FIX
      NP = NNN(2) + NNN(3)                                              STAT
      IST = NNN(1) + 1                                                  STAT
      PRINT 1100,IP2(2)                                                 STAT
C     CALL  KRUSWAL(X(IST),ID1(IST),ID2(IST),NP,0,0,2,NNN(2),LK)        STAT
      IPI=3                                                             FIX2
      CALL KRUSWAL(X(IST),ID1(IST),ID2(IST),NP,0,1,2,NNN(2),LK,IPI)     FIX2
      NCOC = NNN(3)                                                     STAT
      DO 880 J=1,NCOC                                                   STAT
      K = NNN(1) + J                                                    STAT
      L=NNN(1)+NNN(2)+J                                                 FIX
      X(K) = X(L)                                                       STAT
      ID1(K) = ID1(L)                                                   STAT
  880 ID2(K) = ID2(L)                                                   STAT
      NP=NNN(1)+NNN(3)                                                  FIX
      NNN(2)=NNN(3)                                                     FIX
C     CALL  KRUSWAL(X,ID1,ID2,NP,0,0, 2,NNN,LK)                        STAT
  860 PRINT 1100,IP2(3)                                                 DEC1
      IPI=3                                                             FIX2
      IF(ICASE.EQ.4) IPI=2                                              FIX2
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,2,NNN,LK,IPI)                       FIX2
  998 CONTINUE                                                          FIX
      IF(IB.EQ.3) 701,999                                               ANOV
  701 MAXT=MAXB=3                                                       ANOV
      MAXT1=MAXT+1                                                      ANOV
      MAXB1=MAXB+1                                                      ANOV
```

```
      NAMET=4HTIME                                                      ANOV1
      NAMEB=5HASTRO                                                     ANOV1
      IWT=1                                                             ANOV1
      IDEN(1)=KSTUDY                                                    ANOV1
      IDEN(2)=KTEST                                                     ANOV1
      IDEN(8)=3HALL                                                     ANOV1
      IF(ICASE.EQ.2) IDEN(8)=IP2(1)                                     ANOV1
      IF(ICASE.EQ.4) IDEN(8)=IP2(3)                                     ANOV1
      DO 703 J=1,JTOT                                                   ANOV1
      DO 704 I=1,ITOT                                                   ANOV1
      KK=N(I,J)                                                         ANOV1
      PRINT 705,(Y(1,J,K),K=1,KK)                                       ANOV1
  705 FORMAT(F12.4)                                                     ANOV1
  704 CONTINUE                                                          ANOV1
  703 CONTINUE                                                          ANOV1
      PRINT 1000,JOB(3)                                                 ANOV1
      CALL ANOV2NE(MAXT,MAXB,ITOT,JTOT,A,B,AP,Q,N,W,V,WW,VV,Y,NSUMI,    ANOV1
     1  SUMYI,NSUMJ,SUMYJ,SUMYIJ,TAUHAT,BETHAT,THAT,BHAT,SUMIJ,MAXT1,   ANOV1
     2   MAXB1)                                                         ANOV1
      IF(IWT.LT.0) GO TO 999                                            ANOV1
      ALPHA=.05                                                         ANOV1
  797 FORMAT(/////@0 RANKED MEAN CONTRASTS(S-METHOD) ON @,A8,           ANOV1
     2        @ ASSUMING NO INTERACTION@/                              ANOV1
     3        @ AT THE @,F5.3,@ SIGNIFICANCE LEVEL @///)               ANOV1
  798 FORMAT(@1@,8A10)                                                  ANOV1
  799 FORMAT(/////@0 RANKED MEAN CONTRASTS(S-METHOD) ON @,A8,           ANOV1
     2        @ AT THE @,F5.3,@ SIGNIFICANCE LEVEL @///)               ANOV1
      WRITE(6,798) IDEN                                                 ANOV1
      WRITE(6,797) NAMET,ALPHA                                          ANOV1
      CALL SCHEFE(ITOT,TAUHAT,WT,ITHT,ALPHA,KOE,WW,TWOAVT)              ANOV1
      WRITE(6,799) NAMET,ALPHA                                          ANOV1
      CALL SCHEFE(ITOT,THAT,WT,ITHT,ALPHA,KOE,WW,TWOWT)                 ANOV1
      WRITE(6,797) NAMEB,ALPHA                                          ANOV1
      CALL SCHEFE(JTOT,BETHAT,WT,ITHT,ALPHA,KOE,VV,TWOAVB)              ANOV1
      WRITE(6,799) NAMEB,ALPHA                                          ANOV1
      CALL SCHEFE(JTOT,BHAT,WT,ITHT,ALPHA,KOE,VV,TWOWT)                 ANOV1
  999 PRINT 1000,JOB(3)                                                 DEC10
      RETURN                                                            DEC10
  200 FORMAT(10X,F12.4,2X,2A1)                                          STAT
  300 FORMAT(13H0GRAND MEAN =,F10.3,  8H     SD =,F10.3,7H     N =,I3)  STAT
  400 FORMAT(I10,F12.4,2X,A4,,2X,A2)                                    STAT
  900 FORMAT( 7H0MEAN =,F10.3,  8H     SD =,F10.3,7H     N =,I3//)      STAT
      END                                                               STAT
      SUBROUTINE MESIG(LTEXT,NFIRST,NLAST,SMEAN,SIGMA,NN)               MESIG
      DIMENSION LTEXT(1)                                                MESIG
      NN = 0                                                            MESIG
      SUM = 0.0                                                         MESIG
      SSQ = 0.0                                                         MESIG
      IF(NFIRST-NLAST) 10,10,90                                         APR24
   10 DO 140 I=NFIRST,NLAST                                             MESIG
      NSLOT = 1                                                         MESIG
      CALL OTEST(I,SUM,SSQ,NN,LTEXT)                                    MESIG
  140 CONTINUE                                                          MESIG
      IF(NN.EQ.0) GO TO 90                                              DEC10
      SMEAN = SUM/NN                                                    MESIG
      SIGMA=0.                                                          DEC10
      IF(NN.EQ.1) RETURN                                                DEC10
      IF((NN*SSQ-SUM*SUM).LE.0.) RETURN                                 DEC10
      SIGMA = ((NN*SSQ - SUM*SUM)/(NN*(NN - 1)))**0.5                   MESIG
      RETURN                                                            MESIG
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
   90 SMEAN = 0.0                                                          MESIG
      SIGMA = 0.0                                                          MESIG
      RETURN                                                               MESIG
      END                                                                  MESIG
      SUBROUTINE OTEST(I,SUM,SSQ,NN,LTEXT)                                OTEST
      DIMENSION LTEXT(1)                                                   OTEST
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP           XREDUC
      DATA IZ/4H0.00/, ISS/4HSKIP/, INV/4HINVL/                            OTEST
      DATA IBLANK/4H    /                                                  JAN25
      IF(XDATA(I)) 310,410,510                                             OTEST
  310 IXT = -XDATA(I) + 0.5                                                OTEST
      IF(IXT - 2) 315,312,315                                              OTEST
  312 LTEXT(I) = IZ                                                        OTEST
      NN = NN + 1                                                          OTEST
      RETURN                                                               OTEST
  315 LTEXT(I) = ISS                                                       OTEST
      RETURN                                                               OTEST
  410 LTEXT(I) = INV                                                       OTEST
      RETURN                                                               OTEST
  510 NN = NN + 1                                                          OTEST
      LTEXT(I) = IBLANK                                                    OTEST
      SUM = SUM + XDATA(I)                                                 OTEST
      SSQ = SSQ + XDATA(I)*XDATA(I)                                        OTEST
      RETURN                                                               OTEST
      END                                                                  OTEST
      SUBROUTINE DERTST                                                    DERTST
      RETURN                                                               DERTST
      END                                                                  DERTST
      SUBROUTINE  STPLOT(LOOP,NOBST,NSLOT,IWORD,KSTUDY,KTYPE,KMAN,KTEST,STPLOT
     X SMEAN,SIGMA)                                                        STPLOT
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP           XREDUC
      COMMON/HISBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)              XREDUC
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),      STPLOT
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA              STPLOT
      DIMENSION YDATE(200),XP(3)                                           REDUCE
      COMMON/RG/ DATESC(3,5),IVV(4), SSMEAN(3), SSIGMA(3), NNN(3)          STPLOT
C     REAL*8 MSTUDY,MTYPE,MMAN,MTEST                                       STPLOT
      REAL   MSTUDY,MTYPE,MMAN,MTEST                                       STPLOT
      REAL KSTUDY, KTYPE, KMAN, KTEST, MITS                                STPLOT
      DIMENSION NSMAN(7)                                                   STPLOT
      EQUIVALENCE(YDATE,T)                                                 STPLOT
      DATA NSMAN/4,9,8,0,1,2,3/                                            STPLOT
      IF(NOBST - 1) 911,911,1                                              STPLOT
    1 MFIRST = MDATE(1)                                                    STPLOT
      DO 10 II=1,NOBST                                                     STPLOT
      IF(MDATE(II) - MFIRST) 9,10,10                                       STPLOT
    9 MFIRST = MDATE(II)                                                   STPLOT
   10 YDATE(II) = MDATE(II)                                                STPLOT
      J1 = NOBST + 1                                                       STPLOT
      J2 = NOBST + 2                                                       STPLOT
      CALL SCALE(XDATA, 3.0,NOBST,1)                                       STPLOT
C     XDATA(NOBST + 1) = 0.0                                               STPLOT
      IF(XDATA(J2)) 911,911,11                                             STPLOT
   11 IG = LSTUDY(1)                                                       STPLOT
      YDATE(NOBST + 1) = DATESC(1,IG)                                      STPLOT
      YDATE(NOBST + 2) = 24.0                                              STPLOT
      UP = (DATESC(2,IG) - DATESC(1,IG))/YDATE(NOBST + 2)                  STPLOT
      DOWN=(DATESC(3,IG) - DATESC(1,IG))/YDATE(NOBST + 2)                  STPLOT
      XP(1) = 0.1                                                          STPLOT
      XP(2) = 0.1+ UP                                                      STPLOT
```

```
      XP(3) = 0.1+ DOWN                                                 STPLOT
      NTIC = 3                                                          STPLOT
      CALL TEXT(LSTUDY(1), LTYPE(1), LMAN(1), LTEST(1),                 STPLOT
     X KSTUDY, KTYPE, KMAN, KTEST, NITS)                                STPLOT
      KS = KSKIP - KSKIP/3*3                                            STPLOT
      IF(KS) 208,108,208                                                STPLOT
C 108 CALL PLOT(4.25,10.5,-3)                                           STPLOT
  108 CALL PLOT(8.50,10.5,-3)                                           STPLOT
      FPN = LTYPE(1)                                                    STPLOT
      CALL SYMBOL(0.30,-.4,0.10,KSTUDY    , 0.0, 8)                     STPLOT
      CALL SYMBOL(1.20,-.4,0.10, KTEST    , 0.0, 8)                     STPLOT
      CALL SYMBOL(2.00,-.4,0.10,4HTYPE, 0.0, 4)                         STPLOT
      CALL NUMBER(2.40,-.4,0.10,FPN,0.0,-1)                             STPLOT
      CALL SYMBOL(2.60,-.4,0.10,IWORD,0.0,10)                           STPLOT
  208 CALL PLOT(0.0,-3.5,-3)                                            STPLOT
      CALL AXIS(0.0,0.0,12H JULIAN DATE,                                STPLOT
     1    -12, 4.0,00.0,YDATE(J1),YDATE(J2),-1,4,3)                     STPLOT
C     CALL AXIS (X,Y,BCD,NC,SIZE,THETA,YMIN,DY,NDEC,NLAB,NTIC)          STPLOT
C     CALL AXIS(0.0,0.0,16H      DATA VALUE,16,3.0,  90.0, Y(J1),Y(J2)) STPLOT
      CALL AXIS(0.0,0.0,NITS    , 8,3.0,  90.0, XDATA(J1),              STPLOT
     1 XDATA(J2), -1,  4, 1)                                            STPLOT
      FPN = LMAN(1)                                                     STPLOT
      CALL SYMBOL(0.10,3.01,0.07,KMAN, 0.0, 3)                          STPLOT
      DO 308 JJ=1,3                                                     STPLOT
      KG = 4 - JJ                                                       STPLOT
      CALL NUMBER(XP(KG), 2.89, 0.07, SSMEAN(KG), 0.0, 2)               STPLOT
  308 CALL NUMBER(XP(KG), 2.77, 0.07, SSIGMA(KG), 0.0, 2)               STPLOT
      CALL PLOT(UP,0.0,3)                                               STPLOT
      CALL PLOT(UP,3.0,2)                                               STPLOT
      CALL PLOT(DOWN,3.0,3)                                             STPLOT
      CALL PLOT(DOWN,0.0,2)                                             STPLOT
  708 LL = NSMAN(LMAN(1))                                               STPLOT
      CALL LINE(YDATE,XDATA,NOBST,1, 0,LL)                              STPLOT
  811 CONTINUE                                                          STPLOT
      KSKIP = KSKIP + 1                                                 STPLOT
  911 RETURN                                                            STPLOT
      END                                                               STPLOT
      SUBROUTINE TEXT(LSTUDY,LTYPE,LMAN,LTEST,NSTUDY,NTYPE,NMAN,NTEST,  TEXT
     1 NITS)                                                            TEXT
C     REAL*8 KSTUDY,MSTUDY(6),KTEST,MURN(124),MBLU(24),MITS(124),       TEXT
      REAL   KSTUDY,MSTUDY(5),KTEST,MURN(124),MBLU(24),MITS(124),       TEXT
     1MITSB(24),KTYPE,MTYPE(15),MM,KITS                                 TEXT
      REAL NSTUDY, NTYPE, NMAN, NTEST, KMAN, MMAN(9), NITS              TEXT
      DATA MSTUDY/8HSMEAT   ,8HSKYLAB 2,8HSKYLAB 3,8HSKYLAB 4,8HAPOLLO17TEXT
     1/                                                                 TEXT
      DATA MMAN/4HCDR ,4HPLT ,4HSPT ,4HC1  ,4HC2  ,4HC3  ,4HC4  ,       TEXT
     X4HC5  ,4HC6  /                                                    TEXT
      DATA (MURN(I),I=1,107)                                            TEXT
     X          /8HEPI     ,8HNOREPI  ,8HADH     ,8HHYDRO   ,8HALDO    ,TEXT
     X8H17OH    ,8HOSMO    ,8HNA      ,8HK       ,8HMG      ,8HPO4     ,TEXT
     X8HCA      ,8HCL      ,8HH       ,8HSP.GR.  ,8HCREAT   ,8HURICACID,TEXT
     X8HB       ,8HSI      ,8HFE      ,8HAL      ,8HMO      ,8HCU      ,TEXT
     X8HZN      ,8HTI      ,8HNI      ,8HSR      ,8HCR      ,8HBL      ,TEXT
     X8HMN      ,8HLI      ,8HRB      ,8HPD      ,8HAMD     ,8HETIO    ,TEXT
     X8HDHEA    ,8H11=0 AND,8H11=0ETIO,8H11OH AND,8H11OHETIO,8HTOTAL   ,TEXT
     X8HLYS     ,8HHIS     ,8HNH3     ,8HARG     ,8HHYP     ,8HASP     ,TEXT
     X8HTHR     ,8HSER     ,8HGLU     ,8HPRO     ,8HGLY     ,8HALA     ,TEXT
     X8HCYS/2   ,8HVAL     ,8HMET     ,8HILE     ,8HLEU     ,8HTYR     ,TEXT
     X8HPHE     ,8HHLYS    ,8HGAMMA-AB,8HORN     ,8HETH     ,8HNH3     ,TEXT
     X8HLYS     ,8H1-CH3HIS,8HHIS     ,8H3-CH3HIS,8HANS     ,8HTRY     ,TEXT
```

```
     X8HCRE     ,8HCAR     ,8HARG     ,8HPSER    ,8HPETN    ,8HTYR     ,TEXT
     X8HUREA    ,8HHYP     ,8HASP     ,8HTHR     ,8HSER     ,8HASP NH2 ,TEXT
     X8HGLU NH2 ,8HSAR     ,8HPRO     ,8HGLU     ,8HCIT     ,8HGLC NH2 ,TEXT
     X8HGLY     ,8HALA     ,8HALPHA-AA,8HALPHA-AB,8HVAL     ,8HCYS/2   ,TEXT
     X8HCYT     ,8HMET     ,8HILE     ,8HLEU     ,8HTYR     ,8HPHE     ,TEXT
     X8HB-ALA   ,8HB-AIB   ,8H5HIAA   ,8H        ,8H        ,8H        /JAN25
      DATA (MURN(I),I=108,124)/                                          TEXT
     X8H        ,8H        ,8H        ,8H        ,8H        ,8H        ,TEXT
     X8H        ,8H        ,8HCALORIES,8HPROTEIN ,8HDIET CA ,8HDIET P  ,TEXT
     X8HDIET NA ,8HDIET MG ,8HDIET K  ,8HDIET H2O,8HWEIGHT  /           TEXT
      DATA MBLD /8HHYDRO    ,8HALDO    ,8HHGH     ,8HANGIO   ,8HINSULIN ,TEXT
     X8HT4      ,8HACTH    ,8HADH     ,8HPTH     ,8HCAL     ,8HVIT.D.  ,TEXT
     X8HTSH     ,8HOSMO    ,8HNA      ,8HK       ,8HMG      ,8HPO4     ,TEXT
     X8HCA      ,8HCL      ,8HGLU     ,8HTP      ,8HTESTOS  ,8H        ,JAN25
     X8H        /                                                       TEXT
      DATA (MITS(I),I=1,107)                                             TEXT
     X          /8HUG/TV   ,8HUG/TV   ,8HMU/TV   ,8HUG/TV   ,8HUG/TV   ,TEXT
     X8HMG/TV   ,8HMOSM/TV ,8HMEQ/TV ,8HMEQ/TV ,8HMEQ/TV ,8HMG/TV   ,TEXT
     X8HMEQ/TV  ,8HMEQ/TV  ,8HML0     ,8H        ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,TEXT
     X8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,TEXT
     X8HUG/ML   ,8HUG/ML   ,8HUG/ML   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HTMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMMG/TV  ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMMG/TV  ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMMG/TV  ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,TEXT
     X8HMG/TV   ,8HMG/TV   ,8HMG/TV   ,8H        ,8H        ,8H        /JAN25
      DATA (MITS(I),I=108,124)/                                          TEXT
     X8H        ,8H        ,8H        ,8H        ,8H        ,8H        ,TEXT
     X8H        ,8H        ,8HKCAL    ,8HGRAMS   ,8HMG      ,8HMG      ,TEXT
     X8HMG      ,8HMG      ,8HMG      ,8HML      ,8HLBS.    /          TEXT
      DATA MITSB/8HUG/100ML,8HPG/ML   ,8HNG/ML   ,8HMUG/ML/H,8HUU/ML   ,TEXT
     X8HUG/100ML,8H        ,8H        ,8HPG/ML   ,8HPG/ML   ,8HNG/ML   ,TEXT
     X8H        ,8HMOSM/L  ,8HMEQ/L   ,8HMEQ/L   ,8HMG PC   ,8HMG PC   ,JAN25
     X8HMG PC   ,8HMG PC   ,8HMG PC   ,8HMG PC   ,8HUG/TV   ,8H        ,JAN25
     X8H        /                                                       TEXT
      NSTUDY = MSTUDY(LSTUDY)                                            TEXT
      NMAN = MMAN(LMAN)                                                  TEXT
      IF(LTEST - 400) 10,20,20                                           TEXT
   10 KTEST = MBLD(LTEST - 299)                                          TEXT
      KITS = MITSB(LTEST-299)                                            TEXT
      GO TO 30                                                           TEXT
   20 KTEST = MURN(LTEST - 399)                                          TEXT
      KITS = MITS(LTEST-399)                                             TEXT
   30 NTEST = KTEST                                                      TEXT
      NITS = KITS                                                        TEXT
      RETURN                                                             TEXT
      END                                                                TEXT
      SUBROUTINE KRUSWAL(XIN,ID1,ID2IN,N,NCEN,JCP,K,NO,NC,IP2)           SETWRK
C     THIS PROGRAM CALCULATES  A  GENERALIZED KRUSKAL-WALLIS K-SAMPLE    KRUSWAL
C     TEST. WHEN K=2, THE TEST IS EQUAL TO GEHAN'S GENERALIZED WILCOXON  KRUSWAL
C     TEST. CALCULATIONS FOLLOW THE PROCEDURES GIVEN BY MANTEL (BIOMETRIKRUSWAL
C     MARCH 1967)                                                        KRUSWAL
```

```
C                                                                       KRUSWAL
C     IN CASE OF TWO-SAMPLE, IF THE SAMPLE SIZES ARE DIFFERENT, IT IS   KRUSWAL
C     SUGGESTED TO NAME THE ONE WITH LESS OBSERVATIONS SAMPLE 1.        KRUSWAL
C      CALL NKRUSWAL(X,ID1,ID2,NALL,NC,JOP,NCOND,NUNCEN,NCEN)           KRUSWAL
C                                                                       KRUSWAL
C      X(I),I=1,NALL     ARRAY CONTAINING ALL OBS IN THE SAMPLES       KRUSWAL
C       ID1(I)=1         ITH OBS IS UNCENSORED                         KRUSWAL
C             =0         ITH OBS IS CENSORED                           KRUSWAL
C       ID2(I)=J         ITH OBS IS FROM JTH SAMPLE                    KRUSWAL
C       NALL             TOTAL NO OF OBS                               KRUSWAL
C       NC               TOTAL NO OF CENSORED OBS                      KRUSWAL
C       JOP=1            PRINT INFO                                    KRUSWAL
C          =0            DO NOT PRINT                                  KRUSWAL
C       NCOND            NO OF SAMPLES                                 KRUSWAL
C       NUNCEN(I)        NO OF UNCENSORED OBS IN SAMPLE I              KRUSWAL
C       NCEN(I)          NO OF CENSORED OBS IN SAMPLE I                KRUSWAL
      DIMENSION XIN(1),ID1(1),ID2IN(1),NU(6),NC(6),XY(120),ID2(120)     SETWRK
      DIMENSION R1( 600),R2( 600)                                       KRUSWAL
C                                                                       KRUSWAL
C     MAXIMUM NO. OF SAMPLES = 6.                                       KRUSWAL
C     TOTAL NUMBER OF OBSERVATIONS ALLOWED = 600.                       KRUSWAL
C     K = NO. OF SAMPLES                                                KRUSWAL
C                                                                       KRUSWAL
C      MOVE INCOMMING ARRAYS TO WORK ARRAYS                             SETWRK
      DO 10 I=1,N                                                       SETWRK
      XY(I)=XIN(I)                                                      SETWRK
      ID2(I)=ID2IN(I)                                                   SETWRK
   10 CONTINUE                                                          SETWRK
C                                                                       KRUSWAL
C     ORDER OBS. IN ASCENDING ORDER                                     KRUSWAL
C                                                                       KRUSWAL
      CALL SORT2(XY,N,ID1,ID2)                                          KRUSWAL
C     COMPUTATION OF R1                                                 KRUSWAL
C        STEPS 1 AND 2 : RANK FROM LEFT TO RIGHT, OMITTING RIGHT CENSOREDKRUSWAL
C                        VALUES. ASSIGN NEXT HIGHER RANK TO RIGHT       KRUSWAL
C                        CENSORED VALUES                                KRUSWAL
C                                                                       KRUSWAL
      IRANK=0                                                           KRUSWAL
      DO 90I=1,N                                                        KRUSWAL
      IF (ID1(I).EQ.0) GO TO 101                                        KRUSWAL
      IRANK=IRANK+1                                                     KRUSWAL
      R1(I)=IRANK                                                       KRUSWAL
      GO TO 90                                                          KRUSWAL
  101 R1(I)=IRANK+1                                                     KRUSWAL
   90 CONTINUE                                                          KRUSWAL
C                                                                       KRUSWAL
C        STEP 3 : REDUCE  THE RANK OF TIED OBSERVATIONS TO THE LOWEST   KRUSWAL
C                 RANK FOR THE VALUE                                    KRUSWAL
C                                                                       KRUSWAL
      K1=N-1                                                            KRUSWAL
      L1=1                                                              KRUSWAL
   12 IF (XY(L1).NE.XY(L1+1)) GO TO 11                                  KRUSWAL
      JEMP=ID1(L1)*ID1(L1+1)                                            KRUSWAL
      IF(JEMP .EQ. 0) GO TO 11                                          KRUSWAL
      R1(L1+1)=R1(L1)                                                   KRUSWAL
      IF (L1.EQ.K1) GO TO 13                                            KRUSWAL
      L1=L1+1                                                           KRUSWAL
      GO TO 12                                                          KRUSWAL
   11 IF (L1.EQ.K1) GO TO 13                                            KRUSWAL
      L1=L1+1                                                           KRUSWAL
```

```
      GO TO 12                                                          KRUSWAL
 13   CONTINUE                                                          KRUSWAL
C                                                                       KRUSWAL
C     COMPUTATION OF R2                                                 KRUSWAL
C        STEP 1 : RANK FROM RIGHT TO LEFT                               KRUSWAL
C                                                                       KRUSWAL
      DO 14 I=1,N                                                       KRUSWAL
 14   R2(I)=N-I+1                                                       KRUSWAL
C                                                                       KRUSWAL
C        STEP 2 : REDUCE THE RANK OF TIED OBSERVATIONS TO THE LOWEST RANKKRUSWAL
C                 FOR THE VALUE                                         KRUSWAL
C                                                                       KRUSWAL
      L1=N                                                              KRUSWAL
 22   IF (XY(L1).NE.XY(L1-1)) GO TO 21                                  KRUSWAL
      JEMP=ID1(L1)*ID1(L1-1)                                            KRUSWAL
      IF (JEMP .EQ. 0) GO TO 21                                         KRUSWAL
      R2(L1-1)=R2(L1)                                                   KRUSWAL
      IF (L1.EQ. 2) GO TO 23                                            KRUSWAL
      L1=L1-1                                                           KRUSWAL
      GO TO 22                                                          KRUSWAL
 21   IF (L1.EQ. 2) GO TO 23                                            KRUSWAL
      L1=L1-1                                                           KRUSWAL
      GO TO 22                                                          KRUSWAL
 23   CONTINUE                                                          KRUSWAL
C                                                                       KRUSWAL
C        STEP 3 : REDUCE THE RANK OF RIGHT CENSORED OBSERVATIONS TO UNITYKRUSWAL
C                                                                       KRUSWAL
      IF (NCEN .EQ. 0) GO TO 501                                        KRUSWAL
      DO 24 I=1,N                                                       KRUSWAL
      IF (ID1(I) .EQ. 1) GO TO 24                                       KRUSWAL
      R2(I)=1.                                                          KRUSWAL
 24   CONTINUE                                                          KRUSWAL
C                                                                       KRUSWAL
C     COMPUTE FINAL SCORES -R1(I)                                       KRUSWAL
C                                                                       KRUSWAL
 501  CONTINUE                                                          KRUSWAL
      DO 25 I=1,N                                                       KRUSWAL
 25   R1(I)=R1(I)-R2(I)                                                 KRUSWAL
      IF(JOP.NE.1) GO TO 37                                             KRUSWAL
      PRINT 30                                                          KRUSWAL
 30   FORMAT(1H0,8X,@I@,8X,@OBSERVATIONS@,8X,@SAMPLE@,8X,@SCORES@)      KRUSWAL
      DO 31 I=1,N                                                       KRUSWAL
      IF (ID1(I).EQ.0) GO TO 34                                         KRUSWAL
      PRINT 33, I,XY(I),ID2(I),R1(I)                                    KRUSWAL
      GO TO 31                                                          KRUSWAL
   34 PRINT 35, I,XY(I),ID2(I),R1(I)                                    KRUSWAL
 33   FORMAT(1H ,6X,I3,7X,F8.1,15X,I2,12X,F6.0)                         KRUSWAL
 35   FORMAT(1H ,6X,I3,7X,F8.1,1H+,14X,I2,12X,F6.0)                     KRUSWAL
 31   CONTINUE                                                          KRUSWAL
   37 PRINT 36                                                          KRUSWAL
 36   FORMAT(1H0)                                                       KRUSWAL
      IF (K.GT.2) GO TO 200                                             KRUSWAL
      CALL TWOSPL(R1,ID2,N,NU,NC,IP2)                                   FIX2
      GO TO 1000                                                        KRUSWAL
 200  CALL AKSPL(K,N,R1,NU,NC,ID2)                                      KRUSWAL
 1000 CONTINUE                                                          KRUSWAL
      RETURN                                                            KRUSWAL
      END                                                               KRUSWAL
      SUBROUTINE SORT2(X,N,ID,IC)                                       SORT2
      DIMENSION X(1),ID(1),IC(1)                                        SORT2
```

```
      DO 1 I=1,N                                                        SORT2
      J=N-I+1                                                           SORT2
      JJ=J-1                                                            SORT2
      IF (JJ .LT. 1)GO TO 1                                             SORT2
      DO 2 K=1,JJ                                                       SORT2
      IF (X(K).NE.X(J)) GO TO 3                                         SORT2
      IF (ID(K)-ID(J)) 4,2,2                                            SORT2
3     IF (X(K) .LT. X(J)) GO TO 2                                       SORT2
4     X1=X(J)                                                           SORT2
      ITEM=ID(J)                                                        SORT2
      ICTE=IC(J)                                                        SORT2
      X(J)=X(K)                                                         SORT2
      ID(J)=ID(K)                                                       SORT2
      IC(J)=IC(K)                                                       SORT2
      X(K)=X1                                                           SORT2
      ID(K)=ITEM                                                        SORT2
      IC(K)=ICTE                                                        SORT2
2     CONTINUE                                                          SORT2
1     CONTINUE                                                          SORT2
      RETURN                                                            SORT2
      END                                                               SORT2
      SUBROUTINE TWOSPL(R1,ID2,N,NU,NC,IP2)                             FIX2
      DIMENSION ID2(1),R1(1),NU(1),NC(1)                                TWOSPL
C     DATA ONE,TWO/4HONE ,4HTWO /                                       TWOSPL
      DATA ONE/10HONE         /                                         TWOSPL
      DATA TWO/10HTWO         /                                         TWOSPL
      WW=0.                                                             TWOSPL
      DO 26 I=1,N                                                       TWOSPL
      IF(ID2(I).EQ.IP2) GO TO 26                                        FIX2
      WW=WW+R1(I)                                                       TWOSPL
26    CONTINUE                                                          TWOSPL
      SUM=0.                                                            TWOSPL
      DO 27 I=1,N                                                       TWOSPL
27    SUM=SUM+R1(I)**2                                                  TWOSPL
      XN1=NU(1)+NC(1)                                                   TWOSPL
      XN2=NU(2)+NC(2)                                                   TWOSPL
      XN=N                                                              TWOSPL
      VAR=XN1*XN2*SUM/(XN*(XN-1.))                                      TWOSPL
      VAR=SQRT(VAR)                                                     TWOSPL
C                                                                       TWOSPL
C     CALCULATE CONTINUITY CORRECTION.                                  TWOSPL
C                                                                       TWOSPL
      SIGN=-1.                                                          TWOSPL
      IF (WW.LT.0.) SIGN=1.                                             TWOSPL
      COR=.5                                                            TWOSPL
      IF (FLOAT((NC(1)+NC(2))/N) .LT. 0.2) COR=1.0                      TWOSPL
      COR=SIGN*COR                                                      TWOSPL
      IF (WW.EQ.0.) COR=0.                                              TWOSPL
C                                                                       TWOSPL
C     CALCULATE FINAL W SCORE AND PROBABILITY.                          TWOSPL
C                                                                       TWOSPL
      WSCORE=(WW+COR)/VAR                                               TWOSPL
      WSC=WSCORE                                                        TWOSPL
      IF (ABS(WSCORE)-3.1) 305,305,302                                  TWOSPL
305   P1=100.*(1.-PROB(WSC))                                            TWOSPL
      GO TO 301                                                         TWOSPL
302   P1=0.1                                                            TWOSPL
301   CONTINUE                                                          TWOSPL
      P2=P1*2.                                                          TWOSPL
      P2=AMIN1(P2,100.)                                                 TWOSPL
```

```
      PRINT 600, WW,VAR,WSCORE                                          TWOSPL
  600 FORMAT(1H0,3X,1HW,7X,8HST. DEV.,3X,10HASYMPTOTIC/25X,6HWSCORE/1X,FTWOSPL
     19.0,F8.2,7X,F5.2)                                                 DEC10
      PRINT 601, WSCORE,P2,TWO                                          TWOSPL
      PRINT 601, WSCORE,P1,ONE                                          TWOSPL
  601 FORMAT(1X,F5.2,1X,@IS SIGNIFICANT AT THE @, F7.1,@ PERCENT LEVEL -TWOSPL
     1@,A3,@ TAILED TEST@)                                              TWOSPL
      PRINT 602                                                         TWOSPL
  602 FORMAT(1H0)                                                       TWOSPL
      RETURN                                                            TWOSPL
      END                                                               TWOSPL
      SUBROUTINE AKSPL(K,N,R1,NU,NC,ID2)                                AKSPL
      DIMENSION R1(1),NU(1),NC(1),W(6),ID2(1)                           AKSPL
C                                                                       AKSPL
C     TEST STATISTIC FOR K(GREATER THAN 2)-SAMPLE CASE                  AKSPL
C                                                                       AKSPL
      DO 201 I=1,K                                                      AKSPL
  201 W(I)=0.                                                           AKSPL
      DO 202 IJ=1,N                                                     AKSPL
      I=ID2(IJ)                                                         AKSPL
  202 W(I)=W(I)+R1(IJ)                                                  AKSPL
      T=0.                                                              AKSPL
      DO 203 IJ=1,N                                                     AKSPL
  203 T=T+R1(IJ)**2                                                     AKSPL
      B=0.                                                              AKSPL
      PRINT 300                                                         AKSPL
  300 FORMAT(1H0,@SAMPLE@,8X,@W(I)@,10X,@NU(I)@)                        AKSPL
      DO 204 I=1,K                                                      AKSPL
      UC=NU(I)+NC(I)                                                    AKSPL
      PRINT 301, I,W(I),UC                                              AKSPL
  301 FORMAT(1H ,3X,I1,7X,F8.0,7X,F5.0)                                 AKSPL
  204 B=B+(W(I)**2/UC)                                                  AKSPL
      PRINT 302, B,T                                                    AKSPL
  302 FORMAT(1H0,@B =@,F12.2,5X,@T =@,F10.0)                            AKSPL
      WSCORE=(B/T)*FLOAT(N-1)                                           AKSPL
C                                                                       AKSPL
C     WSCORE HAS CHI-SQUARE DISTRIBUTION WITH (K-1) D. F.               AKSPL
C                                                                       AKSPL
      WS=WSCORE                                                         AKSPL
      XK=K-1                                                            AKSPL
      XM=WS/2.                                                          AKSPL
      IC=XK/2.                                                          AKSPL
      SWS=SQRT(WS)                                                      AKSPL
      EM=1./EXP(XM)                                                     AKSPL
      PRINT 205, WSCORE                                                 AKSPL
  205 FORMAT(1H0,@WSCORE = @,F7.3)                                      AKSPL
      XK2=XK/2.                                                         AKSPL
      IF ((XK2-FLOAT(IC)).NE.0) GO TO 500                               AKSPL
      SUM=0.                                                            AKSPL
      PROD=1.                                                           AKSPL
      DO 1 I=1,IC                                                       AKSPL
      IF (I.GT.1) GO TO 2                                               AKSPL
      XI=1                                                              AKSPL
      GO TO 3                                                           AKSPL
    2 XI=I-1                                                            AKSPL
    3 PROD=XI*PROD                                                      AKSPL
      TERM=XM**(I-1)/PROD                                               AKSPL
    1 SUM=SUM+TERM                                                      AKSPL
      CPROB=SUM*EM                                                      AKSPL
      P=100.*CPROB                                                      AKSPL
```

```
      IF (P.LT.0.1000) GO TO 10                                          AKSPL
      PRINT 206, WSCORE,P                                                AKSPL
  206 FORMAT(1H0,F7.3,@ IS SIGNIFICANT AT THE @,F5.1,@ PERCENT LEVEL@)   AKSPL
      GO TO 1000                                                         AKSPL
   10 P=0.10                                                             AKSPL
      PRINT 12, WSCORE,P                                                 AKSPL
   12 FORMAT(1H0,F7.3,@ IS SIGNIFICANT WITH PROBABILITY LESS THAN @,F7.3AKSPL
     1,@ PERCENT LEVEL@)                                                 AKSPL
      GO TO 1000                                                         AKSPL
  500 GAMH=1.7724538509                                                  AKSPL
      TERM=SQRT(XM)/(0.5*GAMH)                                           AKSPL
      SUM=TERM                                                           AKSPL
      NR=(XK-3.)/2.                                                      AKSPL
      IF(NR.EQ.0) GO TO 502                                              AKSPL
      DO 501 I=1,NR                                                      AKSPL
      XI=I                                                               AKSPL
      TERM=TERM*2.*XM/(2.*XI+1.)                                         AKSPL
  501 SUM=SUM+TERM                                                       AKSPL
  502 CHISQ=(SUM*EM)*100.                                                AKSPL
      P2=100.*2.*(1.-PROB(SQS))                                          AKSPL
      P=P2+CHISQ                                                         AKSPL
      IF (P.LT.0.0005) GO TO 14                                          AKSPL
      PRINT 206,  WSCORE,P                                               AKSPL
      GO TO 1000                                                         AKSPL
   14 P=0.001                                                            AKSPL
      PRINT 12,  WSCORE,P                                                AKSPL
 1000 CONTINUE                                                           AKSPL
      RETURN                                                             AKSPL
      END                                                                AKSPL
      FUNCTION PROB(X)                                                   PROB
C                                                                        PROB
C       THIS FUNCTION ROUTINE COMPUTES                                   PROB
C                    DISTRIBUTION FUNCTION(X)        IF X GE 0           PROB
C                1 - DISTRIBUTION FUNCTION(X)        IF X LT 0           PROB
C       OF A ST. NORMAL VARTABLE USING APPROXIMATION 26.2.19  P.932      PROB
C       HANDBOOK OF MATH. FUNCTIONS                                      PROB
      DATA D1,D2,D3,D4,D5,D6/.0498673470,.0211410061,.0032776263,.00038PROB
     10036,.0000488906,.0000053830/                                      PROB
      IF (X) 20,30,40                                                    PROB
   30 PROB=0.5                                                           PROB
      RETURN                                                             PROB
   20 X1=-X                                                              PROB
      GO TO 50                                                           PROB
   40 X1=X                                                               PROB
   50 A=1.+X1*(D1+X1*(D2+X1*(D3+X1*(D4+X1*(D5+X1*D6)))))                 PROB
      PROB=1.-0.5*A**(-16)                                               PROB
   60 CONTINUE                                                           PROB
      RETURN                                                             PROB
      END                                                                PROB
      SUBROUTINE FFOUT(S1, S2, N1, N2,FDAT, PRB)                         FFOUT
      FDAT = (S1/N1)/(S2/N2)                                             FFOUT
      PRB= FISH(FDAT,N1,N2)                                              FFOUT
      PRB = (1.0 - PRB)*100.                                             FFOUT
      IF(PRB - 0.1) 1,2,2                                                FFOUT
    1 PRB = 0.1                                                          FFOUT
    2 RETURN                                                             FFOUT
      END                                                                FFOUT
      FUNCTION FISH(F,N1,N2)                                             FISH
      LOGICAL E1,E2,E3                                                   FISH
       IF(N1.GE.100.AND.N2.GE.100) GOTO 9                                FISH
```

```
C-------------------------------------------------------------------FISH
C     INITIALIZATION AND SETTING OF LOGICAL SWITCHES  TO .TRUE. IF    FISH
C     THE DEGREES OF FREEDOM ARE EVEN                                 FISH
C-------------------------------------------------------------------FISH
      E1=.FALSE.                                                      FISH
      E2=.FALSE.                                                      FISH
      E3=.FALSE.                                                      FISH
      IF(MOD(N1,2).EQ.0) E1=.TRUE.                                    FISH
      IF(MOD(N2,2).EQ.0) E2=.TRUE.                                    FISH
      X=N2/(N2+N1*F)                                                  FISH
      IF(.NOT.(E1.OR.E2)) GO TO 5                                     FISH
      IF(E1.AND..NOT.E2) GO TO 1                                      FISH
      IF(.NOT.E1.AND.E2) GO TO 2                                      FISH
      IF(N1.LE.N2) GO TO 1                                            FISH
C-------------------------------------------------------------------FISH
C   INITIALIZATION FOR SECOND DEGREE OF FREEDOM EVEN  AND LESS THAN   FISH
C      FIRST DEGREE OF FREEDOM IF IT TOO IS EVEN                      FISH
C-------------------------------------------------------------------FISH
    2 I=N1                                                            FISH
      N1=N2                                                           FISH
      N2=I                                                            FISH
      X=1.0-X                                                         FISH
      E3=.TRUE.                                                       FISH
C-------------------------------------------------------------------FISH
C     INITIALIZATION FOR FIRST DEGREE OF FREEDOM EVEN AND LESS THAN   FISH
C   SECOND DEGREE OF FREEDOM IF IT IS EVEN                            FISH
C-------------------------------------------------------------------FISH
    1 Y=1.0-X                                                         FISH
C-------------------------------------------------------------------FISH
C     CALCULATION OF PROBABILITY FOR AT LEAST ONE DEGREE OF FREEDOM EVENFISH
C-------------------------------------------------------------------FISH
      FISH=0.0                                                        FISH
      H=SQRT(X**N2)                                                   FISH
      M=N1/2-1                                                        FISH
      MCDC = M + 1                                                    FISH
      DO 3 ICDC=1,MCDC                                                FISH
      I = ICDC - 1                                                    FISH
      FISH=FISH+H                                                     FISH
    3 H=(H*Y*(N2+2.*I))/(2.*(I+1.))                                   FISH
      IF(E3) GO TO 4                                                  FISH
C-------------------------------------------------------------------FISH
C       ADJUST CALCULATED PROBABILITY IF ITS ONES COMPLEMENT WAS      FISH
C    CALCULATED ORIGINALLY                                            FISH
C-------------------------------------------------------------------FISH
      FISH=1.0-FISH                                                   FISH
      RETURN                                                          FISH
    4 I=N1                                                            FISH
      N1=N2                                                           FISH
      N2=I                                                            FISH
      RETURN                                                          FISH
C-------------------------------------------------------------------FISH
C     CALCULATION OF THE PROBABILITY FOR BOTH DEGREES OF FREEDOM ODD  FISH
C-------------------------------------------------------------------FISH
    5 Y=1.0-X                                                         FISH
      H=.63661977*SQRT(X*Y)                                           FISH
      FISH=.63661977*ACOS(SQRT(X))                                    FISH
      IF(N2.EQ.1) GO TO 8                                             FISH
      M=N2-2                                                          FISH
      DO  6  I=1,M,2                                                  FISH
      FISH=FISH+H                                                     FISH
```

```
    6 H=H*X*(I+1)/(I+2)                                                    FISH
    8 IF(N1.EQ.1) RETURN                                                   FISH
      H=H*N2                                                               FISH
      M=N1-2                                                               FISH
      DO  7  I=1,M,2                                                       FISH
      FISH=FISH-H                                                          FISH
    7 H=H*Y*(N2+I)/(I+2)                                                   FISH
      RETURN                                                               FISH
    9  D1=N1                                                              FISH
       D2=N2                                                               FISH
       DT=(D1/D2)*F                                                        FISH
       DN=SQRT((2.*D2-1.)*DT)-SQRT(2.*D1-1.)                               FISH
       X=DN/SQRT(1.+DT)                                                    FISH
       FISH=PHI(X)                                                         FISH
       RETURN                                                              FISH
      END                                                                  FISH
      REAL FUNCTION PHI(X)                                                 PHI
C-------------------------------------------------------------------------PHI
C                                                                          PHI
C  PHI CALCULATES THE AREA UNDER THE NORMAL CURVE                          PHI
C  A TRANSFORMATION AND J-FRACTION ARE USED ( SEE METHOD )                 PHI
C                                                                          PHI
C-------------------------------------------------------------------------PHI
      LOGICAL UPPER                                                        PHI
      IF(X.LT.-13.27) GO TO 6                                              PHI
      IF(X.GT.8.5) GO TO 8                                                 PHI
      IF(X.NE.0.0) GO TO 2                                                 PHI
      PHI=0.50                                                             PHI
      RETURN                                                               PHI
    2 UPPER=X.GT.0.0                                                       PHI
      Z=    (ABS(X))                                                       PHI
      Y= 5.641895302730 2E-1* EXP(- Z*Z /2.E0)                             PHI
      Z = Z /1.414213562373 1E0                                            PHI
      T=0.0E0                                                              PHI
      IF( ABS(Y/Z).GT.0.0) T=Y/(Z-6.918367561873 0E-6                      PHI
     1+5.002535090039 0E-1/(Z+1.238679761140 9E-2+7.726730086587 8E-1      PHI
     2/(Z-4.326398214305 3E0 +7.345628771805 5E1/(Z+1.504067136429 0E1     PHI
     3+6.208624565723 56E0 /(Z+8.897161213079 1E0 +4.918217184587 4E1      PHI
     4/(Z-2.510823006 9509E0 -2.822597294273 7E0/(Z-9.759791730847 2E-1    PHI
     5+2.424421352683 7E1 /(Z+4.800857012508 1E0 +4.922785391900 2E-1      PHI
     6/(Z+7.662117092766 1E0 +5.028561912578 8E1/(Z-4.652928498465 5E0     PHI
     7  ))))))))))                                                        PHI
      T=T/2.E0                                                             PHI
      IF(UPPER) GO TO 4                                                    PHI
      PHI=T                                                                PHI
      RETURN                                                               PHI
    4 PHI=1.0E0-T                                                          PHI
      RETURN                                                               PHI
    6 PHI = 0.0                                                            PHI
      RETURN                                                               PHI
    8 PHI = 1.0                                                            PHI
      RETURN                                                               PHI
      END                                                                  PHI
      SUBROUTINE ERROR                                                     ERROR
      COMMON /SISBUF/ IBUF(35),IUR(125),IBL(35),IBUFF(30),KEE,KEF,         ERROR
     +                MFUNC,IFILE,IERR                                     ERROR
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)              ERROR
      GO TO(100,200,300,400) IFILE                                         ERROR
  100 IERR = IFETCH(FITRA1,3LIPS)                                          ERROR
      CALL STOREF(FITRA1,3LIPS,0)                                          ERROR
```

```
      GO TO 500                                                          ERROR
  200 IERR = IFETCH(FITRA2,3LIRS)                                        ERROR
      CALL STOREF(FITRA2,3LIRS,0)                                        ERROR
      GO TO 500                                                          ERROR
  300 IERR = IFETCH(FITRA3,3LIRS)                                        ERROR
      CALL STOREF(FITRA3,3LIRS,0)                                        ERROR
      GO TO 500                                                          ERROR
  400 IERR = IFETCH(FITRA4,3LIRS)                                        ERROR
      CALL STOREF(FITRA4,3LIRS,0)                                        ERROR
  500 IF(IERR.EQ.0) RETURN                                               ERROR
      PRINT 601,IFILE,IERR,IEE,KEF                                       ERROR
      RETURN                                                             ERROR
  601 FORMAT(* RA*,I3,* ERROR *,O10,2I10)                                ERROR
      END                                                                ERROR
      SUBROUTINE ANOV2NF (MAXT,MAXB,ITOT,JTOT,A,B,P,Q,N,W,V,WW,VV,Y,     ANOV
     2                    NSUMI, SUMYI,NSUMJ,SUMYJ,SUMYIJ,TAUHAT,BETHAT,THAT,  ANOV
     3                    BHAT,SUMIJ,MAXT1,MAXB1)                        ANOV
C                                                                        ANOV
C         TWO-WAY ANALYSIS OF VARIANCE WITH UNEQUAL NUMBERS PER CELL     ANOV
C                                                                        ANOV
C         MODEL                                                          ANOV
C                                                                        ANOV
C     Y(I,J,K) = MU + TAU(I) + BETA(J) + INTERACTION(I,J) + ERROR(I,J,K) ANOV
C                                                                        ANOV
C                                                                        ANOV
C         MODEL ASSUMING ADDITIVITY                                      ANOV
C                                                                        ANOV
C     Y(I,J,K) = MU + TAU(I) + BETA(J) + ERROR(I,J,K)                    ANOV
C                                                                        ANOV
C                                                                        ANOV
C         SUBROUTINE ARGUMENTS                                           ANOV
C                                                                        ANOV
C IN  MAXT        MAXIMUM FIRST DIMENSION OF Y,A,N,SUMYIJ                ANOV
C IN  MAXB        MAXIMUM SECOND DIMENSION OF Y,FIRST OF B               ANOV
C IN  ITOT        NUMBER OF TAU EFFECTS                                  ANOV
C IN  JTOT        NUMBER OF BETA EFFECTS                                 ANOV
C     A           MAXT BY MAXT WORK ARRAY --- TAU EFFECTS  A*TAUHAT = Q  ANOV
C     B           MAXB BY MAXP WORK ARRAY ---BETA EFFECTS  B*BETHAT = P  ANOV
C     P           MAXT WORK VECTOR                                       ANOV
C     Q           MAXB WORK VECTOR                                       ANOV
C IN  N           MAXT BY MAXB (ACTUAL ITOT BY JTOT) NUMBERS OF SAMPLES  ANOV
C                                 PER CELL                               ANOV
C     W           ITOT WEIGHTS OF TAU EFFECTS  (SEE IWT IN COMMON)       ANOV
C     V           JTOT WEIGHTS OF BETA EFFECTS (SEE IWT IN COMMON)       ANOV
C     VW          MAX(ITOT,JTOT) WORK VECTOR FOR WEIGHTS                 ANOV
C IN  Y           ITOT BY JTOT BY N(I,J) ARRAY OF DATA                   ANOV
C     NSUMI       JTOT VECTOR OF SUMS OF N(I,J) -- SUMED ON I            ANOV
C     SUMYI       JTOT VECTOR OF SUMS OF Y(I,J,K) -- SUMED ON K AND I    ANOV
C     NSUMJ       ITOT VECTOR OF SUMS OF N(I,J) -- SUMED ON J            ANOV
C     SUMYJ       ITOT VECTOR OF SUMS OF Y(I,J,K) -- SUMED ON K AND J    ANOV
C     SUMYIJ      ITOT BY JTOT ARRAY OF SUM OF Y(I,J,K) -- SUMED ON K    ANOV
C     TAUHAT      ITOT VECTOR OF ESTIMATES OF TAU EFFECTS                ANOV
C     BETHAT      JTOT VECTOR OF ESTIMATES OF BETA EFFECTS               ANOV
C     X1          MAX(ITOT,JTOT) WORK VECTOR                             ANOV
C         COMMON BLOCK AOV2NF ARGUMENTS                                  ANOV
C IN  IDEN        10 LENGTH VECTOR CONTAINING AOV TABLE TITLE (40 ALPHA) ANOV
C IN  NAMET       2 LENGTH VECTOR CONTAINING THE NAME OF THE TAU EFFECTS ANOV
C IN  NAMEB       2 LENGTH VECTOR CONTAINING THE NAME OF THE BETA EFFECTSANOV
C IN  IWT         IS 1, IF WEIGHTS W,V ARE TO BE 1/ITOT AND 1/JTOT, RESP.ANOV
C                    2, IF W,V AZE TO BE 1/NSUMJ(I),1/NSUMI(J),RESP.     ANOV
```

```
C                3. IF W,V CONTAIN,ON CALL TO ANOVA2NE, THE WEIGHTS   ANOV
C                   TO BE USED                                        ANOV
C                                                                     ANOV
C     THE REMAINING COMMON BLOCK VARIABLES ARE VARIABLES CALCULATED IN ANOV
C        ONE OF THE TWO ANALYSIS OF VARIANCE TABLES PREPARED BY ANOVA2NANOV
C                                                                     ANOV
C     AOV TABLE ONE CALCULATES WITH THE TAU EFFECTS ADJUSTED FOR UNEQUALANOV
C        CELL NUMBERS.  MNSQT,MNSQE,MNSST ARE TYPED REAL AND NOT INTEGEANOV
C                                                                     ANOV
C       S.V.              D.F.        S.S.        M.S.        F-RATIO ANOV
C                                                                     ANOV
C       MU,BETA           JTOT        SUMSQX                          ANOV
C                                                                     ANOV
C       TAU (ADJ.)        ITOT-1      TAU         MNSQT       FRATIO  ANOV
C       NO INTERACTION                                                ANOV
C                                                                     ANOV
C       ERROR             IERR        ERROR       MNSQE               ANOV
C       NO INTERACTION                                                ANOV
C                                                                     ANOV
C       TAU (ADJ.)        ITOT-1      SST         MNSST       FSST    ANOV
C                                                                     ANOV
C       INTERACTION       INT         TIN         TINT        FINT    ANOV
C       EFFECT                                                        ANOV
C                                                                     ANOV
C       WITHIN CELL       ITHT        WITHT       WIT                 ANOV
C       ERROR                                                         ANOV
C                                                                     ANOV
C       TOTAL             NSUM        SUMSQY                          ANOV
C                                                                     ANOV
C                                                                     ANOV
C     AOV TABLE TWO CALCULATES WITH THE BETA EFFECTS ADJUSTED FOR     ANOV
C        UNEQUAL CELL NUMBERS.  MNSQB,MNSQBE,MNSSB ARE TYPED REAL     ANOV
C                                                                     ANOV
C       S.V.              D.F.        S.S.        M.S.        F-RATIO ANOV
C                                                                     ANOV
C       MU,TAU            ITOT        SUMSQT                          ANOV
C                                                                     ANOV
C       BETA (ADJ.)       JTOT-1      BETA        MNSQB       FRATB   ANOV
C       NO INTERACTION                                                ANOV
C                                                                     ANOV
C       ERROR             IERR        ERRORB      MNSQBE              ANOV
C       NO INTERACTION                                                ANOV
C                                                                     ANOV
C       BETA (ADJ.)       JTOT-1      SSB         MNSSB       FSSB    ANOV
C                                                                     ANOV
C       INTERACTION       INT         BIN         BINT        FINB    ANOV
C       EFFECT                                                        ANOV
C                                                                     ANOV
C       WITHIN CELL       ITHT        WITHT       WIT                 ANOV
C       ERROR                                                         ANOV
C       TOTAL             NSUM        SUMSQY                          ANOV
C                                                                     ANOV
C                                                                     ANOV
      DIMENSION Y(MAXT,MAXB,1),N(MAXT,1),NSUMI(1),NSUMJ(1),SUMYI(1)   ANOV
     2         ,Q(1),A(MAXT1,1),TAUHAT(1),X1(10,1),SUMYJ(1)           ANOV
      DIMENSION SUMYIJ(MAXT,1),BETHAT(1),P(1),B(MAXB1,1)              ANOV
      DIMENSION  W(1), V(1), WW(1), VV(1)                             ANOV
      DIMENSION IDEN(8)                                               ANOV
      DIMENSION  THAT(1), BHAT(1)                                     ANOV
      REAL MNSSB                                                      ANOV
```

```
      REAL MNSST                                                        ANOV
      REAL MNSQB,MNSQBE                                                 ANOV
      REAL MNSQT, MNSQE                                                 ANOV
      DIMENSION INIT(37)                                                ANOV
      EQUIVALENCE (INIT(1),SUMSQX)                                      ANOV
      COMMON / AOV2NE /  IDEN,NAMET,NAMEB,IWT,SUMSQX,TAU,MNSQT,FRATIO,  ANOV
     2                   IERR,ERROR,MNSQE,SST,MNSST,FSST,INT,TIN,TINT,  ANOV
     3                   FINT,ITHT,WITHT,WIT,NSUM,SUMSQY,SUMSQT,BETA,   ANOV
     4                   MNSQB,FRATB,ERRORB,MNSQBE,SSB,MNSSB,FSSB,BIN,  ANOV
     5                   BINT,FINB,PFRAT,PFSST,PFINT,PFRATB,PFSSB,PFINB ANOV
      DO 4 I=1,37                                                       ANOV
    4 INIT(I)=0                                                         ANOV
 1008 FORMAT(////10X, A8,@ --- WEIGHTS@/(1X,6G20.7))                    ANOV
      GO TO (210,220,240),IWT                                           ANOV
  210 WT=1./FLOAT(ITOT)                                                 ANOV
C                                                                       ANOV
C         EQUAL WEIGHTS                                                 ANOV
C                                                                       ANOV
      DO 211 I=1,ITOT                                                   ANOV
  211 W(I)=WT                                                           ANOV
      WT=1./FLOAT(JTOT)                                                 ANOV
      DO 212 J=1,JTOT                                                   ANOV
  212 V(J)=WT                                                           ANOV
      GO TO 240                                                         ANOV
C                                                                       ANOV
C         PROPORTIONAL WEIGHTS                                          ANOV
C                                                                       ANOV
  220 SSS=0.                                                            ANOV
      DO 222 I=1,ITOT                                                   ANOV
      WT=0.                                                             ANOV
      DO 221 J=1,JTOT                                                   ANOV
  221 WT=WT+N(I,J)                                                      ANOV
      W(I)=1./WT                                                        ANOV
  222 SSS=SSS+W(I)                                                      ANOV
      SSS=1./SSS                                                        ANOV
      DO 225 I=1,ITOT                                                   ANOV
  225 W(I)=W(I)*SSS                                                     ANOV
      SSS=0.                                                            ANOV
      DO 224 J=1,JTOT                                                   ANOV
      WT=0.                                                             ANOV
      DO 223 I=1,ITOT                                                   ANOV
  223 WT=WT+N(I,J)                                                      ANOV
      V(J)=1./WT                                                        ANOV
  224 SSS=SSS+V(J)                                                      ANOV
      SSS=1./SSS                                                        ANOV
      DO 226 J=1,JTOT                                                   ANOV
  226 V(J)=V(J)*SSS                                                     ANOV
      GO TO 240                                                         ANOV
  240 WRITE(6,1008) NAMET,(W(I),I=1,ITOT)                               ANOV
      WRITE(6,1008) NAMEB,(V(J),J=1,JTOT)                               ANOV
  250 NSUM=0                                                            ANOV
      SUMIJ=0.                                                          ANOV
      SUMSQY = 0.0                                                      ANOV
C                                                                       ANOV
C          FORM SUM OF SQUARES AND SUM OF N                             ANOV
C               AND SUM OF N AND OF Y OVER I                            ANOV
      DO 1 J = 1, JTOT                                                  ANOV
      NSUMI(J) = 0                                                      ANOV
      SUMYI(J) = 0.0                                                    ANOV
C                                                                       ANOV
```

```
      DO 1 I = 1, ITOT                                                  ANOV
      NSUM = NSUM + N(I, J)                                             ANOV
      NSUMI(J) = NSUMI(J) + N(I, J)                                     ANOV
      SUMYIJ(I,J)=0.                                                    ANOV
      K = N(I, J)                                                       ANOV
      DO 1  IK = 1, K                                                   ANOV
      SUMSQY = SUMSQY + Y(I, J, IK) * Y(I, J, IK)                       ANOV
      SUMYI(J) = SUMYI(J) + Y(I, J, IK)                                 ANOV
    1 CONTINUE                                                          ANOV
      IJTOT=ITOT*JTOT                                                   ANOV
      INT=IJTOT-ITOT-JTOT+1                                             ANOV
      ITHT=NSUM-IJTOT                                                   ANOV
      ITOTP1 = ITOT  + 1                                                ANOV
      JTOTP1 = JTOT + 1                                                 ANOV
      ITOT1=ITOT-1                                                      ANOV
      JTOT1=JTOT-1                                                      ANOV
      IERR=NSUM-ITOT-JTOT+1                                             ANOV
C                                                                       ANOV
C     CALCULATE AOV TABLE FOR ADJUSTED TAU EFFECTS                      ANOV
C                                                                       ANOV
C         FORM SUM OF N OVER J AND BEGIN TO FORM Q TO BE USED           ANOV
C         IN SOLUTION FOR TAU                                           ANOV
C                                                                       ANOV
      DO 2 I = 1, ITOT                                                  ANOV
      NSUMJ(I) = 0                                                      ANOV
      SUMYJ(I) = 0.0                                                    ANOV
      Q(I) = 0.0                                                        ANOV
      DO 2 J = 1, JTOT                                                  ANOV
      NSUMJ(I) = NSUMJ(I) + N(I, J)                                     ANOV
      Q(I)=Q(I) + FLOAT(N(I,J)) * SUMYI(J)/FLOAT(NSUMI(J))              ANOV
      K = N(I, J)                                                       ANOV
      DO 11  IK = 1, K                                                  ANOV
      SUMYIJ(I,J)=SUMYIJ(I,J) + Y(I,J,IK)                               ANOV
   11 SUMYJ(I) = SUMYJ(I) + Y(I, J, IK)                                 ANOV
      SUMIJ=SUMIJ + SUMYIJ(I,J)                                         ANOV
      SUMYIJ(I,J)=SUMYIJ(I,J)/FLOAT(K)                                  ANOV
    2 CONTINUE                                                          ANOV
      SUMIJ=SUMIJ/FLOAT(NSUM)                                           ANOV
      WRITE(6,3000) SUMIJ                                               ANOV
      WRITE(6,3003) NAMET                                               ANOV
      WRITE(6,3001) (SUMYJ(I),I=1,ITOT)                                 ANOV
      WRITE(6,3004) NAMET                                               ANOV
      WRITE(6,3005) (NSUMJ(I),I=1,ITOT)                                 ANOV
      WRITE(6,3003) NAMEB                                               ANOV
      WRITE(6,3001) (SUMYI(J),J=1,JTOT)                                 ANOV
      WRITE(6,3004) NAMEB                                               ANOV
      WRITE(6,3005) (NSUMI(J),J=1,JTOT)                                 ANOV
      WRITE(6,3002)                                                     ANOV
      DO 500 I=1,ITOT                                                   ANOV
      WRITE(6,3001) (SUMYIJ(I,J),J=1,JTOT)                              ANOV
  500 CONTINUE                                                          ANOV
      WRITE(6,3006)                                                     ANOV
      DO 501 I=1,ITOT                                                   ANOV
      WRITE(6,3005) (N(I,J),J=1,JTOT)                                   ANOV
  501 CONTINUE                                                          ANOV
 3000 FORMAT(@0MEAN EST @,G15.7)                                        ANOV
 3001 FORMAT(1X,10G12.4)                                                ANOV
 3002 FORMAT(@0CELL MEANS@)                                             ANOV
 3003 FORMAT(@0@, A8,@ SUMS@)                                           ANOV
 3004 FORMAT(@0@, A8,@ NUMBER SUMS@)                                    ANOV
```

```
 3005 FORMAT(1X,10I12)                                                  ANOV
 3006 FORMAT(@0CELL NUMBERS@)                                           ANOV
C                                                                       ANOV
C         CALCULATE WITHIN SUMS OF SQUARES                              ANOV
C                                                                       ANOV
      WITHT=0.                                                          ANOV
      DO 12 I=1,ITOT                                                    ANOV
      DO 12 J=1,JTOT                                                    ANOV
      K=N(I,J)                                                          ANOV
      DO 12 IK=1,K                                                      ANOV
   12 WITHT=WITHT + (Y(I,J,IK)-SUMYIJ(I,J))**2                          ANOV
      IF(ITHT.LE.0) GO TO 15                                            ANOV
      IF(WITHT.GT.0.) GO TO 112                                         ANOV
      WRITE(6,113) WITHT                                                ANOV
  113 FORMAT(@0 ***** WITHIN CELL ERROR IS @,G15.6,@ ... PROCESSING ABOR ANOV
     2TED ***** @)                                                      ANOV
      IWT=-1                                                            ANOV
      RETURN                                                            ANOV
  112 WIT=WITHT/FLOAT(ITHT)                                             ANOV
C                                                                       ANOV
C         FORM THE MATRIX A AND AUGMENT IT TO ASTAR, A NONSINGULAR      ANOV
C         MATRIX IN THE FOLLOWING MANNER                                ANOV
C                                                                       ANOV
C                              A(ITOT X ITOT)     1 (ITOT X 1)          ANOV
C                   ASTAR =                                             ANOV
C                              1-(1 X ITOT)       0 (1 X 1)            ANOV
C                                                                       ANOV
   15 DO 3 IR = 1, ITOT                                                 ANOV
      DO 3 IS = 1, ITOT                                                 ANOV
      A(IR, IS) = 0.0                                                   ANOV
      DO 5 J = 1, JTOT                                                  ANOV
    5 A(IR,IS) = A(IR,IS) + FLOAT(N(IR,J)*N(IS,J)) / FLOAT(NSUMI(J))    ANOV
      A(IR, IS) = -A(IR, IS)                                            ANOV
      IF(IR.EQ.IS) A(IR,IS)=FLOAT(NSUMJ(IR))+A(IR,IS)                   ANOV
    3 CONTINUE                                                          ANOV
C                                                                       ANOV
C                                                                       ANOV
      DO 7 I = 1, ITOT                                                  ANOV
      A(ITOTP1, I) = 1.0                                                ANOV
      A(I, ITOTP1) = 1.0                                                ANOV
C                                                                       ANOV
C         FINISH CALCULATION OF Q                                       ANOV
      Q(I) = SUMYJ(I) - Q(I)                                            ANOV
      X1(I,1)=Q(I)                                                      ANOV
    7 CONTINUE                                                          ANOV
C                                                                       ANOV
C         STORE 0 IN A(NTOTP1, NTOTP1)                                  ANOV
C                                                                       ANOV
      A(ITOTP1, ITOTP1) = 0.0                                           ANOV
C                                                                       ANOV
C         CALL MATRIX INVERSION SUBROUTINE ON A                         ANOV
C                                                                       ANOV
      CALL MATINV(A, ITOTP1, X1,  0, DET,MAXT1)                         ANOV
C                                                                       ANOV
C         CALCULATE TAU                                                 ANOV
C                                                                       ANOV
      DO 8 I = 1, ITOT                                                  ANOV
      TAUHAT(I) = 0.0                                                   ANOV
      DO 8 J = 1, ITOT                                                  ANOV
    8 TAUHAT(I) = TAUHAT(I) + A(I, J)*Q(J)                              ANOV
```

```
C                                                                       ANOV
      TAU = 0.0                                                         ANOV
      DO 9 I = 1, ITOT                                                  ANOV
    9 TAU = TAU + TAUHAT(I)* Q(I)                                       ANOV
C                                                                       ANOV
C         CALCULATE SUM OF SQUARES DUE TO MU AND BETA, UNADJUSTED       ANOV
C                                                                       ANOV
      SUMSQX = 0.0                                                      ANOV
      DO 10 J = 1, JTOT                                                 ANOV
   10 SUMSQX = SUMSQX + SUMYT(J) * SUMYI(J) /FLOAT(NSUMI(J))            ANOV
C                                                                       ANOV
C         AND MEAN SQUARE DUE TO TAU                                    ANOV
      MNSQT = TAU/FLOAT(ITOT1)                                          ANOV
C                                                                       ANOV
C         CALCULATE SUM OF SQUARES DUE TO ERROR                         ANOV
      ERROR = SUMSQY - SUMSQX - TAU                                     ANOV
      IF(ERROR.GT.0) GO TO 100                                          ANOV
      WRITE(6,101) ERROR                                                ANOV
  101 FORMAT(@0 ***** ERROR SUM OF SQUARES IS @,G15.6,@ ... PROCESSING AANOV
     2BORTED ***** @)                                                   ANOV
      IWT=-2                                                            ANOV
      RETURN                                                            ANOV
  100 MNSQE=ERROR/FLOAT(IERR)                                           ANOV
C                                                                       ANOV
C         FROM WHICH AN F-RATIO WITH                                    ANOV
C                 (ITOT -1) AND (NSUM - JTOT -ITOT + 1)                 ANOV
C         DEGREES OF FREEDOM                                            ANOV
C                                                                       ANOV
      FRATIO = MNSQT / MNSQE                                            ANOV
      PFRAT = FISH(FRATIO,ITOT1,IERR)                                   ANOV
      WRITE(6,2009)IDEN                                                 ANOV
      WRITE (6, 2002)                                                   ANOV
      WRITE(6,2004) NAMEB,JTOT,SUMSQX                                   ANOV
      WRITE(6,2005) NAMET,ITOT1,TAU,MNSQT,FRATIO,PFRAT                  ANOV
      WRITE (6, 2006) IERR, ERROR, MNSQE                                ANOV
      SSS=0.                                                            ANOV
      SST=0.                                                            ANOV
      WT=0.                                                             ANOV
      DO 14 I=1,ITOT                                                    ANOV
      SSR=0.                                                            ANOV
      WW(I)=0.                                                          ANOV
      DO 13 J=1,JTOT                                                    ANOV
      SSR=SSR + V(J)*SUMYIJ(I,J)                                        ANOV
   13 WW(I)=WW(I) + V(J)*V(J)/FLOAT(N(I,J))                             ANOV
      WW(I)=1./WW(I)                                                    ANOV
      WT=WT + WW(I)                                                     ANOV
      SSS=SSS + WW(I)*SSR                                               ANOV
      SST=SST + WW(I)*SSR*SSR                                           ANOV
   14 THAT(I)=SSR-SUMIJ                                                 ANOV
      SST=SST - SSS*SSS/WT                                              ANOV
      MNSST=SST/FLOAT(ITOT1)                                            ANOV
      IF(ITHT.LE.0) GO TO 16                                            ANOV
      FSST=MNSST/WIT                                                    ANOV
      PFSST = FISH(FSST,ITOT1,ITHT)                                     ANOV
   16 WRITE(6,2014) NAMET,ITOT1,SST,MNSST,FSST,PFSST                    ANOV
      TIN=ERROR-WITHT                                                   ANOV
      TINT=TIN/FLOAT(INT)                                               ANOV
      IF(ITHT.LE.0) GO TO 17                                            ANOV
      FINT=TINT/WIT                                                     ANOV
      PFINT = FISH(FINT,INT,ITHT)                                       ANOV
```

```
   17 WRITE(6,2012) INT,TIN,TINT,FINT,PFINT                              ANOV
      WRITE(6,2013) ITHT,WITHT,WIT                                       ANOV
      WRITE (6, 2003) NSUM, SUMSQY                                       ANOV
      WRITE(6,2016) NAMET,(TAUHAT(I),I=1,ITOT)                           ANO
      WRITE(6,2011) NAMET,(THAT(I),I=1,ITOT)                             AN
      WRITE(6,2010) NAMET,DET                                            ANO
C                                                                        ANO
C     CALCULATE AOV TABLE FOR ADJUSTED BETA EFFECTS                      ANO
C                                                                        ANO
      DO 53 IR= 1,JTOT                                                   ANO
      DO 53 IS = 1,JTOT                                                  ANO
      B(IR,IS) = 0.                                                      ANO
      DO 55 J= 1, ITOT                                                   ANO
   55 B(IR,IS) = B(IR,IS) + FLOAT(N(J,IR)*N(J,IS)) / FLOAT(NSUMJ(J))     ANO
      B(IR,IS) = -B(IR,IS)                                               ANO
      IF(IR.EQ.IS) B(IR,IS) = FLOAT(NSUMI(IR)) + B(IR,IS)                ANO
   53 CONTINUE                                                           ANO
      DO 52 J= 1,JTOT                                                    ANO
      P(J) = 0.                                                          ANO
      DO 52 I = 1,ITOT                                                   ANO
      P(J) = P(J) + FLOAT(N(I,J)) * SUMYJ(I)/FLOAT(NSUMJ(I))             ANO
   52 CONTINUE                                                           ANO
      DO 57 J=1,JTOT                                                     ANO
      B(JTOTP1,J) = 1.                                                   ANO
      B(J,JTOTP1) = 1.                                                   ANO
C                                                                        ANO
C          FINISH CALCULATION OF P                                       ANO
      P(J) = SUMYI(J) - P(J)                                             ANO
      X1(J,1)=P(J)                                                       ANO
   57 CONTINUE                                                           ANO
      B(JTOTP1,JTOTP1) = 0.                                              AND
C                                                                        ANO
C        CALL MATRIX INVERSION SUBROUTINE ON B                           ANO
C                                                                        ANO
      CALL MATINV(B,JTOTP1,X1,0,DET,MAXB1)                               ANO
C                                                                        ANO
C        CALCULATE BETA                                                  ANO
C                                                                        ANO
      DO 58 I = 1,JTOT                                                   ANO
      BETHAT(I) = 0.                                                     ANO
      DO 58 J = 1,JTOT                                                   ANO
   58 BETHAT(I) = BETHAT(I) + B(I,J)*P(J)                                ANO
C                                                                        ANO
      BETA = 0.                                                          ANO
      DO 59 J = 1,JTOT                                                   ANO
   59 BETA = BETA + BETHAT(J)*P(J)                                       ANO
C                                                                        ANO
C        CALCULATE SUM OF SQUARES DUE TO MU AND TAU,UNADJUSTED           ANO
C                                                                        ANO
      SUMSQT = 0.                                                        ANO
      DO 60 I = 1,ITOT                                                   ANO
   60 SUMSQT = SUMSQT + SUMYJ(I) * SUMYJ(I) /FLOAT(NSUMJ(I))            ANO
C                                                                        ANO
C        CALCULATE SUM OF SQUARES DUE TO ERROR                           ANO
      ERRORB = SUMSQY - SUMSQT - BETA                                    ANO
C                                                                        ANO
C        AND MEAN SQUARE DUE TO BETA                                     ANO
      MNSQB = BETA/FLOAT(JTOT1)                                          ANO
C                                                                        ANOV
C        AND DUE TO ERROR                                                ANOV
```

```
      IF(ERRORB.GT.0.) GO TO 160                                        ANOV
      WRITE(6,101) ERRORB                                               ANOV
      IWT=-3                                                            ANOV
      RETURN                                                            ANOV
  160 MNSQBE=ERRORB/FLOAT(IERR)                                         ANOV
C                                                                       ANOV
C        AND F-RATIO                                                    ANOV
      FRATB = MNSQB  / MNSQBE                                           ANOV
      PFRATB = FISH(FRATB,JTOT1,IERR)                                   ANOV
      WRITE(6,2009)IDEN                                                 ANOV
      WRITE(6,2002)                                                     ANOV
      WRITE(6,2004) NAMET,ITOT,SUMSQT                                   ANOV
      WRITE(6,2005) NAMEB,JTOT1,BETA,MNSQB,FRATB,PFRATB                 ANOV
      WRITE(6,2006) IERR,ERRORB,MNSQBE                                  ANOV
      WT=0.                                                             ANOV
      SSB=0.                                                            ANOV
      SSS=0.                                                            ANOV
      DO 64 J=1,JTOT                                                    ANOV
      SSR=0.                                                            ANOV
      VV(J)=0.                                                          ANOV
      DO 63 I=1,ITOT                                                    ANOV
      SSR=SSR + SUMYIJ(I,J)*W(I)                                        ANOV
   63 VV(J)=VV(J) + W(I)*W(I)/FLOAT(N(I,J))                             ANOV
      VV(J)=1./VV(J)                                                    ANOV
      WT=WT + VV(J)                                                     ANOV
      SSS=SSS + VV(J)*SSR                                               ANOV
      SSB=SSB + VV(J)*SSR*SSR                                           ANOV
   64 BHAT(J)=SSR-SUMIJ                                                 ANOV
      SSB=SSB-SSS*SSS/WT                                                ANOV
      MNSSB=SSB/FLOAT(JTOT1)                                            ANOV
      IF(ITHT.LE.0) GO TO 66                                            ANOV
      FSSB=MNSSB/WIT                                                    ANOV
      PFSSB = FISH(FSSB,JTOT1,ITHT)                                     ANOV
   66 WRITE(6,2014) NAMEB,JTOT1,SSB,MNSSB,FSSB,PFSSB                    ANOV
      BIN=ERRORB-WITHT                                                  ANOV
      BINT=BIN/FLOAT(INT)                                               ANOV
      IF(ITHT.LE.0) GO TO 65                                            ANOV
      FINB=BINT/WIT                                                     ANOV
      PFINB = FISH(FINB,INT,ITHT)                                       ANOV
   65 WRITE(6,2012) INT,BIN,BINT,FINB,PFINB                             ANOV
      WRITE(6,2013) ITHT,WITHT,WIT                                      ANOV
      WRITE(6,2003) NSUM,SUMSQY                                         ANOV
      WRITE(6,2016) NAMEB,(BETHAT(J),J=1,JTOT)                          ANOV
      WRITE(6,2011) NAMEB,(BHAT(J),J=1,JTOT)                            ANOV
      WRITE(6,2010) NAMEB,DET                                           ANOV
      RETURN                                                            ANOV
 2002 FORMAT(@0@,40X,@UNBALANCED TWO-WAY ANALYSIS OF VARIANCE@///       ANOV
     21H , 18X, 2HSV, 19X, 2HDF, 19X, 2HSS, 19X, 2HMS, 20X, 1HF,       ANOV
     317X,@PROB F@)                                                     ANOV
 2003 FORMAT (1H0, 10X, 5HTOTAL, 20X, I5, 15X, G15.7)                   ANOV
 2004 FORMAT(1H0,10X,@MU, @, A8,13X,I5,15X,G15.7/11X,@(UNADJUSTED)@)    ANOV
 2005 FORMAT(1H0,10X, A8,@ (ADJ.)@,10X,I5,15X,3(G15.7,5X),G15.7         ANOV
     2       /11X,@ASSUME ADDITIVE@)                                    ANOV
 2006 FORMAT(1H0,10X,@ERROR@,20X,I5,15X,2(G15.7,5X)/11X,@ADDITIVE@)     ANOV
 2009 FORMAT(1H1 /////////// 30X,8A10)                                   ANOV
 2010 FORMAT (/////,1X,A8,@ EFFECT MATRIX DETERMINANT = @,G15.7)        ANOV
 2011 FORMAT(////10X,@ESTIMATES OF @, A8,@  EFFECTS@/(1X,6G20.7))       ANOV
 2012 FORMAT(1H0,10X,@INTERACTION@,14X,I5,15X,3(G15.7,5X),G15.7)        ANOV
 2013 FORMAT(1H0,10X,@WITHIN CELL ERROR@,8X,I5,15X,2(G15.7,5X))         ANOV
 2014 FORMAT(1H0,10X, A8,@(ADJ.)@,11X,I5,15X,3(G15.7,5X),G15.7          ANOV
```

```
     2       /11X,@(WEIGHTED)@)                                              ANOV
 2016 FORMAT(////10X,@ESTIMATES OF @, A8,@  EFFECTS ASSUMING NO INTERAC[ANOV
     2ION@/(1X,6G20.7))                                                      ANOV
      END                                                                    ANOV
      SUBROUTINE MATINV(A,N,B,M,DETERM,MAX)                                  MA
      DIMENSION IPIVOT(21), A(MAX, 10), B(10, 1), INDEX(21, 2),              MATV
     2          PIVOT(21)                                                   MATV
      EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM), (AMAX, T, SWAP)              MATV
C     INITIALIZATION                                                         MATV
   10 DETERM=1.0                                                             MATV
   15 DO 20 J=1,N                                                            MATV
   20 IPIVOT(J)=0                                                            MATV
   30 DO 550 I=1,N                                                           MATV
C     SEARCH FOR PIVOT ELEMENT                                               MATV
   40 AMAX=0.0                                                               MATV
   45 DO 105 J=1,N                                                           MATV
   50 IF (IPIVOT(J)-1) 60, 105, 60                                           MATV
   60 DO 100 K=1,N                                                           MATV
   70 IF (IPIVOT(K)-1) 80, 100, 740                                          MATV
   80 IF(ABS(AMAX)-ABS(A(J,K)))85,100,100                                    MATV
   85 IROW=J                                                                 MATV
   90 ICOLUM=K                                                               MATV
   95 AMAX=A(J,K)                                                            MATV
  100 CONTINUE                                                               MATV
  105 CONTINUE                                                               MATV
  110 IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                                        MATV
C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL                      MATV
  130 IF (IROW-ICOLUM) 140, 260, 140                                         MATV
  140 DETERM=-DETERM                                                         MATV
  150 DO 200 L=1,N                                                           MATV
  160 SWAP=A(IROW,L)                                                         MA
  170 A(IROW,L)=A(ICOLUM,L)                                                  MATV
  200 A(ICOLUM,L)=SWAP                                                       MATV
  205 IF(M) 260, 260, 210                                                    MATV
  210 DO 250 L=1, M                                                          MATV
  220 SWAP=B(IROW,L)                                                         MATV
  230 B(IROW,L)=B(ICOLUM,L)                                                  MATV
  250 B(ICOLUM,L)=SWAP                                                       MATV
  260 INDEX(I,1)=IROW                                                        MATV
  270 INDEX(I,2)=ICOLUM                                                      MATV
  310 PIVOT(I)=A(ICOLUM,ICOLUM)                                              MATV
  320 DETERM=DETERM*PIVOT(I)                                                 MATV
C     DIVIDE PIVOT ROW BY PIVOT ELEMENT                                      MATV
  330 A(ICOLUM,ICOLUM)=1.0                                                   MATV
  340 DO 350 L=1,N                                                           MATV
  350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)                                       MATV
  355 IF(M) 380, 380, 360                                                    MATV
  360 DO 370 L=1,M                                                           MATV
  370 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT(I)                                       MATV
C     REDUCE NON-PIVOT ROWS                                                  MATV
  380 DO 550 L1=1,N                                                          MATV
  390 IF(L1-ICOLUM) 400, 550, 400                                            MATV
  400 T=A(L1,ICOLUM)                                                         MATV
  420 A(L1,ICOLUM)=0.0                                                       MATV
  430 DO 450 L=1,N                                                           MATV
  450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T                                          MATV
  455 IF(M) 550, 550, 460                                                    MA
  460 DO 500 L=1,M                                                           MATV
  500 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T                                          MATV
  550 CONTINUE                                                               MATV
```

```
C      INTERCHANGE COLUMNS                                              MATV
  600 DO 710 I=1,N                                                      MATV
  610 L=N+1-I                                                           MATV
  620 IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630                          MATV
  630 JROW = INDEX(L,1)                                                 MATV
  640 JCOLUM=INDEX(L,2)                                                 MATV
  650 DO 705 K=1,N                                                      MATV
  660 SWAP=A(K,JROW)                                                    MATV
  670 A(K,JROW)=A(K,JCOLUM)                                             MATV
  700 A(K,JCOLUM)=SWAP                                                  MATV
  705 CONTINUE                                                          MATV
  710 CONTINUE                                                          MATV
  740 RETURN                                                            MATV
      END                                                               MATV
      FUNCTION TWOAVT(EMS,JSUB,N2,ALPHA,W,I1,IS)                        FUNCTS
      COMMON DUM(357),Y(729),A(4,4),B(4,4)                              FUNCTS
      TWOAVT=SQRT(EMS*JSUB*FISHIN(ALPHA,JSUB,N2)*(A(II,II)-2.*A(II,IS)  FUNCTS
     2            +A(IS,IS)))                                           FUNCTS
      RETURN                                                            FUNCTS
      END                                                               FUNCTS
      FUNCTION TWOAVB(EMS,JSUB,N2,ALPHA,W,II,IS)                        FUNCTS
      COMMON DUM(357),Y(729),A(4,4),B(4,4)                              FUNCTS
      TWOAVB=SQRT(EMS*JSUB*FISHIN(ALPHA,JSUB,N2)*(B(II,II)-2.*B(II,IS)  FUNCTS
     2            +B(IS,IS)))                                           FUNCTS
      RETURN                                                            FUNCTS
      END                                                               FUNCTS
      FUNCTION TWOWT (EMS,JSUB,N2,ALPHA,W,II,IS)                        FUNCTS
      DIMENSION W(1)                                                    FUNCTS
      TWOWT=SQRT(EMS*JSUB*FISHIN(ALPHA,JSUB,N2)*(1./W(II) + 1./W(IS)))  FUNCTS
      RETURN                                                            FUNCTS
      END                                                               FUNCTS
      SUBROUTINE SCHEFE(K,TMEANS,EMS,N2,ALPHA,KOE,W,FDUM)               SCHEFE
C                                                                       SCHEFE
C     SCHEFFE'S CONTRASTS FOR SIGNIFICANT DIFFERENCES IN MEANS          SCHEFE
C                                                                       SCHEFE
C  K=NUMBER OF MEANS TO BE TESTED   TMEANS=VECTOR OF MEANS TO BE TESTED SCHEFE
C  EMS=ERROR MEAN SQUARE (DENOMINATOR OF F TEST)                        SCHEFE
C  N2 =ERROR DEGREES OF FREEDOM                                         SCHEFE
C     FDUM IS A DUMMY EXTERNALLY DEFINED FUNCTION TO CALCULATE CONTRAST SCHEFE
C FOR STANDARD OUTPUT (K.LE.10) ALL MEANS HAVING THE SAME LINE UNDER THESCHEFE
C     STATISTACALLY THE SAME.                                           SCHEFE
      DIMENSION TMEANS(K),PRINT(20),KOE(1),W(1)                         SCHEFE
      DATA BLANK /6H       /,XLINE / 6HXXXXXX /                         SCHEFE
      ISAVE2=0                                                          SCHEFE
C RANK MEANS FROM LOW TO HIGH                                           SCHEFE
      CALL ORDERM(TMEANS,K,KOE)                                         SCHEFE
      WRITE(6,51) (KOE(I),I=1,K)                                        SCHEFE
   51 FORMAT(@ @,10(6X,I2,4X))                                          SCHEFE
      WRITE(6,53)(TMEANS(I),I=1,K)                                      SCHEFE
 53   FORMAT(1X ,10G12.6)                                               SCHEFE
C DO COMPARISONS                                                        SCHEFE
      ISTOP=K                                                           SCHEFE
 25   ISM1=ISTOP-1                                                      SCHEFE
      DO 24 I=1,20                                                      SCHEFE
 24   PRINT(I)=BLANK                                                    SCHEFE
      DO 30 I=1,ISM1                                                    SCHEFE
      ISAVE=I                                                           SCHEFE
      JSUB=ISM1-I+1                                                     SCHEFE
 29   IF(ISAVE2.EQ.I    )GO TO 31                                       SCHEFE
      RANGE = TMEANS(ISTOP)-TMEANS(I)                                   SCHEFE
```

```
      II=KOE(I)                                                          SCHEFE
      IS=KOE(ISTOP)                                                      SCHEFE
      S=FDUM(EMS,JSUB,N2,ALPHA,W,II,IS)                                  SCHEFE
      IF(RANGE-S) 34,34,30                                               SCHEFE
30    CONTINUE                                                           SCHEFE
31    ISTOP=ISTOP-1                                                      SCHEFE
      IF(ISTOP.EQ.1)GO TO 99                                             SCHEFE
      GO TO 25                                                           SCHEFE
34    IF(K.LE.10)GO TO 35                                                SCHEFE
      WRITE(6,33)ALPHA,(TMEANS(KKK),KKK=ISAVE,ISTOP)                     SCHEFE
33    FORMAT( @0 THE FOLLOWING MEANS ARE STATISTICALLY THE SAME AT THE @SCHEFE
     2       ,F5.3,@ ALPHA LEVEL@,/,(10G12.6))                          SCHEFE
      GO TO 38                                                           SCHEFE
35    DO 37 I=ISAVE,ISTOP                                                SCHEFE
      II=2*I                                                             SCHEFE
      PRINT(II-1) = XLINE                                                SCHEFE
   37 PRINT(II) = XLINE                                                  SCHEFE
      WRITE(6,52)(PRINT(KK),KK=1,20)                                     SCHEFE
38    IF(ISAVE.EQ.1)GO TO 99                                             SCHEFE
      ISAVE2=ISAVE                                                       SCHEFE
      GO TO 31                                                           SCHEFE
99    RETURN                                                             SCHEFE
52    FORMAT(@ @,20A6)                                                   SCHEFE
      END                                                                SCHEFE
       SUBROUTINE ORDERM(X,N,KOE)                                        ORDER
       DIMENSION X(N)  ,KOE(N)                                           ORDER
      NN=N                                                               ORDER
       K1=2                                                              ORDER
      DO 80 I=1,NN                                                       ORDER
   80 KOE(I)=I                                                           ORDER
    4  DO 99 I=K1,NN                                                     ORDER
       IF(X(I).LT.X(I-1)) GOTO 76                                        ORDER
   99 CONTINUE                                                           ORDER
      RETURN                                                             ORDER
   76  DO 82 K= 1,NN                                                     ORDER
      IF(X(I).LT.X(K)) GO TO 84                                          ORDER
   82 CONTINUE                                                           ORDER
   84 Z=X(I)                                                             ORDER
      KK=KOE(I)                                                          ORDER
      I1=I-1                                                             ORDER
      DO 86KJ=K,I1                                                       ORDER
      J=K+I1-KJ                                                          ORDER
      KOE(J+1)=KOE(J)                                                    ORDER
   86 X(J+1)=X(J)                                                        ORDER
      X(K)=Z                                                             ORDER
      KOE(K)=KK                                                          ORDER
      K1=I+1                                                             ORDER
       IF(K1.GT.NN) RETURN                                               ORDER
       GOTO 4                                                            ORDER
       END                                                               ORDER
       FUNCTION FISHIN(ALPHA,N1,N2)                                      FISHIN
C-------------------------------------------------------------------------FISHIN
C CALCULATES THE INVERSE @F@ VALUE GIVEN THE CONFIDENCE COEFFICIENT       FISHIN
C   ALPHA AND THE DEGREES OF FREEDOM(N).                                 FISHIN
C-------------------------------------------------------------------------FISHIN
      Y1=N1                                                              FISHIN
      Y2=N2                                                              FISHIN
C-------------------------------------------------------------------------FISHIN
C  ADJUST FOR DEGREES OF FREEDOM EQUAL TO 1                              FISHIN
C-------------------------------------------------------------------------FISHIN
```

```
      IF(N1.EQ.1)  Y1=2                                                 FISHIN
      IF(N2.EQ.1)  Y2=2                                                 FISHIN
C-----------------------------------------------------------------------FISHIN
C CALL PHINV TO GET INVERSE NORMAL VALUE OF 1.-ALPHA                    FISHIN
C-----------------------------------------------------------------------FISHIN
       X=PHINV(1.-ALPHA)                                                FISHIN
C-----------------------------------------------------------------------FISHIN
C  COMPUTE LAMDA VALUE                                                  FISHIN
C-----------------------------------------------------------------------FISHIN
      Y=(X**2-3.)/6.                                                    FISHIN
      IC=0                                                              FISHIN
C-----------------------------------------------------------------------FISHIN
C  COMPUTE THE INITIAL APPROXIMATION TO THE INVERSE @F@ FUNCTION        FISHIN
C-----------------------------------------------------------------------FISHIN
      Y1=1./(Y1-1.)                                                     FISHIN
      Y2=1./(Y2-1.)                                                     FISHIN
      H=2./(Y1+Y2)                                                      FISHIN
      X=X*SQRT(H+Y)/H-(Y1-Y2)*(Y+5./6.-2./(3.*H))                       FISHIN
      X=EXP(2.*X)                                                       FISHIN
C-----------------------------------------------------------------------FISHIN
C  COMPUTE THE CONSTANT TO THE @F@ DISTRIBUTION,TESTING FOR N1 AND/OR N2FISHIN
C   ODD OR EVEN.                                                        FISHIN
C-----------------------------------------------------------------------FISHIN
      G=1.                                                              FISHIN
      IB1=2                                                             FISHIN
      IF(MOD(N1,2).EQ.0) GO TO 1                                        FISHIN
      G=1.7724539                                                       FISHIN
      IB1=1                                                             FISHIN
    1 IB2=2                                                             FISHIN
      IF(MOD(N2,2).EQ.0) GO TO 2                                        FISHIN
      G=G*1.7724539                                                     FISHIN
      IB2=1                                                             FISHIN
    2 IB3=2                                                             FISHIN
      IF(MOD(N1+N2,2).EQ.0) GO TO 3                                     FISHIN
      G=G/1.7724539                                                     FISHIN
      IB3=1                                                             FISHIN
    3 IF((IB1+IB2).NE.2)  G=2.*G                                        FISHIN
      IF((N1+N2).LE.3)  GO TO 5                                         FISHIN
      ND=N1+N2-2-IB3                                                    FISHIN
      ND1 = ND + 1                                                      FISHIN
      DO 4 II=1,ND1,2                                                   FISHIN
      I = II - 1                                                        FISHIN
      IF((IB1+I).LE.(N1-2))  G=G*(IB1+I)                                FISHIN
      IF((IB2+I).LE.(N2-2))  G=G*(IB2+I)                                FISHIN
    4 G=G/(IB3+I)                                                       FISHIN
C-----------------------------------------------------------------------FISHIN
C  COMPUTE THE VALUE OF FISHIN                                          FISHIN
C-----------------------------------------------------------------------FISHIN
    5 Y2=N2/(N2+N1*X)                                                   FISHIN
      Y1=1.-Y2                                                          FISHIN
      Y=1.+(G*(1.-ALPHA-FISH(X,N1,N2)))/SQRT(Y1**N1*Y2**N2)             FISHIN
      FISHIN=X*Y                                                        FISHIN
C-----------------------------------------------------------------------FISHIN
C  IF FISHIN IS NEGATIVE, RESET FISHIN TO .5*LAST APPROXIMATION(X).     FISHIN
C-----------------------------------------------------------------------FISHIN
      IF(Y.LT.0.)  FISHIN=.5*X                                          FISHIN
C-----------------------------------------------------------------------FISHIN
C  IF THE ABSOLUTE VALUE OF THE DIFFERENCE IS LESS THAN .5E-6, RETURN   FISHIN
C-----------------------------------------------------------------------FISHIN
      IF(ABS(X/FISHIN-1.).LT.(.5E-6))  GO TO 7                          FISHIN
```

```
C-------------------------------------------------------------------------FISHIN
C  IF THE RELATIVE VALUE OF THE DIFFERENCE IS LESS THAN .5E-6, RETURN    FISHIN
C-------------------------------------------------------------------------FISHIN
      IF(ABS(X-FISHIN).LT.(.5E-6)) GO TO 7                                FISHIN
      IC=IC+1                                                             FISHIN
      IF(IC.GT.100)  RETURN                                               FISHIN
C-------------------------------------------------------------------------FISHIN
C  SET THE APPROXIMATION EQUAL TO FISHIN AND CONTINUE TO ITERATE.         FISHIN
C-------------------------------------------------------------------------FISHIN
      X=FISHIN                                                            FISHIN
      GO TO 5                                                             FISHIN
    7 RETURN                                                              FISHIN
      END                                                                 FISHIN
      FUNCTION PHINV( P )                                                 PHINV
      IF(P .EQ. 1.0)GO TO 98                                              PHINV
      IF(P .EQ. 0.0)GO TO 97                                              PHINV
      IF(P .GT. 1.0)GO TO 88                                              PHINV
      IF(P .LT. 0.0)GO TO 88                                              PHINV
      K = 1                                                               PHINV
      IF(P .GT. 0.5)GO TO 47                                              PHINV
    6 T3=SQRT(-2.0*ALOG(P))                                               PHINV
      T4P=2.515517+.802853*T3+.010328*T3*T3                               PHINV
      T5P=1.0+1.432788*T3+.189269*T3*T3+.001308*T3*T3*T3                  PHINV
      XT=T3-T4P/T5P                                                       PHINV
      XT=-XT                                                              PHINV
   13 DO 53 I=1,100                                                       PHINV
      PHF = EXP(-0.5*XT*XT)                                               PHINV
      PT = PHI (XT)                                                       PHINV
      IF(ABS(P-PT) .LT. P*4.0E-8)GO TO 99                                 PHINV
      Z = (P-PT)*2.50662827 / PHF                                         PHINV
      XT = XT + Z                                                         PHINV
   53 CONTINUE                                                            PHINV
      GO TO 99                                                            PHINV
   47  P = 1.0 - P                                                        PHINV
      K = 2                                                               PHINV
      GO TO 6                                                             PHINV
   99 GO TO (26,27),K                                                     PHINV
   26 PHINV = XT                                                          PHINV
      RETURN                                                              PHINV
   27 PHINV = -XT                                                         PHINV
      P = 1.0 - P                                                         PHINV
      RETURN                                                              PHINV
   98 PHINV = 1.0E+38                                                     PHINV
      RETURN                                                              PHINV
   97 PHINV = -1.0E+38                                                    PHINV
      RETURN                                                              PHINV
   88 WRITE(6,10) P                                                       PHINV
   10 FORMAT(1H0,5X,29HARGUMENT NOT A PROBABILITY = ,5X,E14.7  )          PHINV
      RETURN                                                              PHINV
      END                                                                 PHINV
```

D.7 Control cards and modification deck for modifying retrieval program to perform analysis of covariance

```
NASARET,CM20000,T1000.          NEWTON
ACCOUNT,AN12318,,
RFL,70000.
GET,RTREVE.
GET(ANACSRC)
MODIFY,F,P=RTREVE,LO=CET.
FTN,I,L=0,PL=50000,T.
ATTACH,FTNIMSL/UN=LB12345.
GET,FTNPLT/UN=AN12005.
GET,RA1.
GET,RA2.
GET,RA3.
GET,RA4.
RFL,110000.
MODE(1)
LOAD,LGO,FTNIMSL,FTNPLT.
EXECUTE.
PACK,ZZZZZEF.
COPYSBF,ZZZZZEF,OUTPUT.
EXIT.
PACK,ZZZZZEF.
COPYSBF,ZZZZZEF,OUTPUT.
-
*CREATE  ANACSRC
*IDENT  APR24
*DECK  STAT
*D  75
      IVV(1)=IVV(2)=IVV(3)=0
*D  108
      IF(IFIRST-ILAST)162,163,166
  163 PRINT 401
  401 FORMAT(*  SINGLE VALUE NOT ANALYZED*)
*DECK  MESIG
*D  6
      IF(NFIRST-NLAST) 10,10,90
*IDENT  SETWRK
*DECK  KRUSWAL
*D  1
      SUBROUTINE KRUSWAL(XIN,ID1,ID2IN,N,NCEN,JOP,K,NU,NC,IP2)
*D  22
      DIMENSION XIN(1),ID1(1),ID2IN(1),NU(6),NC(6),XY(120),ID2(120)
*I  28
C     MOVE INCOMMING ARRAYS TO WORK ARRAYS
      DO 10 I=1,N
      XY(I)=XIN(I)
      ID2(I)=ID2IN(I)
   10 CONTINUE
*IDENT  FIX
*DECK RETD
*D  1
      PROGRAM RETD(INPUT,OUTPUT,TAPE6=OUTPUT)
*DECK  STAT
*MODNAME  DEC10
*D  8,19
*D  23
*D  25
*D  26
      IF(ICASE.EQ.2) GO TO 998
*MODNAME STAT
*I 190
```

```
      IF(ICASE.EQ.4) 710,711
  710 IF(NNN(2).EQ.1) GO TO 712
      GO TO 713
  712 NNN1=NNN(1)+1
      NSLTM1=J-1
      DO 179 I=NNN1,NSLTM1
  179 X(I)=X(I+1)
  713 NNN(2)=NNN(3)
      NNN(3)=0
  711 CONTINUE
*I  219
      P(3)=0
*D  222,223
      PRINT 3600,IP2(1),FVAL,PREVAL
 3600 FORMAT(1H0,A8,F8.3,23H IS SIGNIFICANT AT THE    ,F5.1,
*D  230
      PRINT 3600,IP2(2),FVAL,PREVAL
*D  235
      PRINT 3600,IP2(3),FVAL,PREVAL
*D  262
      L=NNN(1)+NNN(2)+J
*I 265
      NP=NNN(1)+NNN(3)
      NNN(2)=NNN(3)
*I  267
  993 CONTINUE
*IDENT FIX2
*DECK  STAT
     155
      NN=NN-NNN(2)
*I  165
      NN=NN-NNN(3)
*D  247
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,NT,NNN,LK,LK)
*D  252
      IPI=2
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,2,NNN,LK,IPI)
*D  257
      IPI=3
      CALL KRUSWAL(X(IST),ID1(IST),ID2(IST),NP,0,1,2,NNN(2),LK,IPI)
*D  267
      IPI=3
      IF(ICASE.EQ.4) IPI=2
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,2,NNN,LK,IPI)
*DECK  KRUSWAL
*D 113
      CALL TWOSPL(R1,ID2,N,NU,NC,IP2)
*DECK  TWOSPL
*D  1
      SUBROUTINE TWOSPL(R1,ID2,N,NU,NC,IP2)
*D  8
      IF(ID2(I).EQ.IP2) GO TO 26
*IDENT  XREDUC
*DECK RETD
*ODNAME  REDUCE
*D  1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D  2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*DECK  RET
```

```
*MODNAME  REDUCE
*D   1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D   2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*D   3
      IF(IH.LE.151)  GO TO 10
*D   4
      IF(NSMP.LE.150)  GO TO 35
*DECK  SETCR1
*MODNAME  REDUCE
*D   1
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*DECK  RETRVE
*MODNAME  REDUCE
*D   1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D   2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*D   3
      IF(N.GT.150)  GO TO 110
*DECK  STAT
*MODNAME  REDUCE
*D   1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D   2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*DECK  OTEST
*MODNAME  REDUCE
*D   1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*DECK  STPLOT
*MODNAME  REDUCE
*D   1
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP
*D   2
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)
*IDENT  ANAC1
*DECK STAT
*I   6
      COMMON CHEM(200),TIME(200),NOB(3),TREAT(6),SL(6),SS(12),XMSC(12)
     2    ,F(12),PROBF(12),XM(6),YM(6),NOFC(12),SXX(6),SYY(6),SXY(6)
      COMMON ZCH(2,60),WORK(2,60)
      DIMENSION C(28)
*D   186,187
      GO TO 321
*I   188
  321 XOUT(J)=MDATE(I)
*I   190
C     SET FLAG FOR ANACOVA
      IF(LMAN(1).EQ.1) IANC=1
      IF(ICASE.EQ.2.OR.ICASE.EQ.3) GO TO 322
      PRINT 605
  605 FORMAT(*  MISSING INFLIGHT DATA - NO ANACOVA POSSIBLE*)
      IANC=0
      GO TO 716
C     PICK INFLIGHT DATA ONLY INTO CHEM AND TIME
C     X IS CHANGED HERE AND LATER
  322 IF(LMAN(1).EQ.1) LOBS=0
      I=NUM(1)+1
```

```
      J=NNN(1)+NNN(2)
      DO 714 K=I,J
      LOBS=LOBS+1
      CHEM(LOBS)=X(K)
      TIME(LOBS)=XOUT(K)
  714 CONTINUE
      NOB(LMAN(1))=NNN(2)
  716 CONTINUE
*MODNAME  FIX
*I  20
      IF(LMAN(1).LT.3.OR.IANC.NE.1) GO TO 999
      PRINT 1000,JOB(3)
      PRINT 604,LSTUDY(1),KSTUDY,IWORD,LTEST(1),KTEST,LTYPE(1)
  604 FORMAT(7HOSTUDY ,I2,2X,A8,2X,A10,2X,5HTEST ,I3,2X,A8,2X,5HTYPE ,
     2I3,*  INFLIGHT DATA FOR ANACOVA*)
      PRINT 602
      K=0
      DO 715 L=1,3
      J=K+1
      K=K+NOB(L)
      PRINT 602
  602 FORMAT(1H )
      M=0
      DO 717 I=J,K
      M=M+1
  717 PRINT 601,M,CHEM(I),TIME(I)
  715 CONTINUE
  601 FORMAT(I4,F12.4,F5.0)
      NT=3
      IOPT=2
      CALL ANACOVA(CHEM,TIME,LOBS,NT,NOB,TREAT,SL,SS,XMSC,F,PROBF,
     2     IOPT,XM,YM,SXX,SYY,SXY,NDFC)
      IF(IOPT.LT.0) PRINT 603
  603 FORMAT(@  ANACOVA  ERROR**********@)
      J=1
      K=0
      DO 719 LAS=1,3
      K=K+NOB(LAS)
      L=0
      DO 718 I=J,K
      L=L+1
      ZCH(2,L) =0.
      ZCH(1,L)=CHEM(I)-(YM(LAS)+SL(LAS)*(TIME(I)-XM(LAS)))
      CHEM(I)=ZCH(1,L)
      WORK(1,L)=0.
      WORK(2,L)=0.
  718 CONTINUE
      CALL LAGCOR(NOB(LAS),CHEM(J),NOB(LAS),CHEM(J),C,1,28)
      PRINT 610,LAS
  610 FORMAT(*0 MAN*,I3,*  LAG CORRELATIONS*/*0SEQ    C-VALUE*)
      PRINT 611,(I,C(I),I=1,28)
  611 FORMAT(I6,G15.6)
      N=NOB(LAS)
      CALL FOURG(ZCH,N,-1,WORK)
      DO 723 I=1,N
      WORK(1,I)=ZCH(1,I)**2+ZCH(2,I)**2
  723 WORK(2,I)=ATAN2(ZCH(2,I),ZCH(1,I))
      PRINT 602
      PRINT 607
  607 FORMAT(1H0,T18,*FAST FOURIER TRANSFORMS*/* SEQ OMEGA*,
```

```
     1        6X,@REAL          IMAGINARY      R**2+I**2      ARCTAN(I/R)@)
      DO 720 M=1,N
      QNM=FLOAT(N)/FLOAT(M-1)
      IF(M.EQ.1) QNM=0.
      PRINT 606,M,QNM,ZCH(1,M),ZCH(2,M),WORK(1,M),WORK(2,M)
  720 CONTINUE
  606 FORMAT(1X,I3,F6.2,4G15.6)
      J=K+1
  719 CONTINUE
-
JOB  DATE    400
STUDYEACH      3
DATE RANGE   189   209
DATE RANGE   220   265
TYPE EACH      1
TEST RANGE   406   409
END
EOF
-
-
```

D.8 Analysis of covariance program as contained in file ANACSRC

```
NASA,CM50000,    L NEWTON
ACCOUNT,AN12318.
COPY(INPUT,ANACSRC)
REPLACE(ANACSRC)
-
ANACOVA
      SUBROUTINE ANACOVA(Y,X,L ,NT,NOB,TREAT,SL,SS,XMS,F,PROBF,IOPT,
     Z               XM,YM,SXX,SYY,SXY,NDF)
      DIMENSION Y(2),X(2),NOB(2),YM(2),XM(2),SL(2),SS(2),XMS(2),F(2),
     2          PROBF(2),SXX(2),SYY(2),SXY(2),TREAT(2),NDF(2)
      SS(1)=0
      SS(9)=0
      NTT=NT+1
      NT2=NT+2
      NT3=NT+3
      SS(12)=0
      L=0
      DO 9 I=1,NT3
      SXX(I)=0
      SYY(I)=0
      SXY(I)=0
      XM(I)=0
      YM(I)=0
    9 CONTINUE
      DO 20 I=1,NT
      IF(NOB(I).EQ.0) GO TO 99
      NO=NOB(I)
      DO 10 J=1,NO
      L=L+1
      XM(I)=XM(I)+X(L)
      YM(I)=YM(I)+Y(L)
      SXX(I)=SXX(I)+X(L)*X(L)
      SYY(I)=SYY(I)+Y(L)*Y(L)
   10 SXY(I)=SXY(I)+X(L)*Y(L)
      YM(NTT)=YM(NTT)+YM(I)
      XM(NTT)=XM(NTT)+XM(I)
      SXX(NTT)=SXX(NTT)+SXX(I)
      SXY(NTT)=SXY(NTT)+SXY(I)
      SYY(NTT)=SYY(NTT)+SYY(I)
      SS(12)=SS(12)+SYY(I)
      XM(I)=XM(I)/NO
      YM(I)=YM(I)/NO
      SXX(I)= SXX(I)-NO*XM(I)**2
      SYY(I)=(SYY(I)-NO*YM(I)**2)/(NO-1.)
      SXY(I)= SXY(I)-NO*XM(I)*YM(I)
      SL(I)=SXY(I)/SXX(I)
      TREAT(I)= YM(I)-SL(I)*XM(I)
      SS(1)=SS(1)+NO*YM(I)*YM(I)
      SS(9)=SS(9)+SXX(I)*SL(I)**2
      SXY(NT3)=SXY(NT3)+NO*XM(I)*YM(I)
      SXX(NT3)=SXX(NT3)+NO*XM(I)**2
      SXX(NT2)=SXX(NT2)+SXX(I)
      SXY(NT2)=SXY(NT2)+SXY(I)
      SXX(I)=SXX(I)/(NO-1.)
      SXY(I)=SXY(I)/(NO-1.)
   20 CONTINUE
      YM(NTT)=YM(NTT)/L
      XM(NTT)=XM(NTT)/L
      SXX(NTT)=(SXX(NTT)-L*XM(NTT)*XM(NTT))/(L-1.)
      SXY(NTT)=(SXY(NTT)-L*XM(NTT)*YM(NTT))/(L-1.)
```

```
      SSU=L*YM(NTT)*YM(NTT)
      SYY(NTT)=(SYY(NTT)-SSU)/(L-1.)
      SL(NTT)=SXY(NTT)/SXX(NTT)
      SS(NTT)=SS(12)
      SL(NT2)=SXY(NT2)/SXX(NT2)
      SS(6)=SXX(NT2)*SL(NT2)**2
      SXX(NT2)=SXX(NT2)/(L-NT)
      SXY(NT2)=SXY(NT2)/(L-NT)
      SS(8)=SS(9)-SS(6)
      SXX(NT3)= (SXX(NT3) - L*XM(NTT)**2) /(NT-1.)
      SXY(NT3)= (SXY(NT3) - L*XM(NTT)*YM(NTT))/(NT-1.)
      IF(SXX(NT3).GT.0) GO TO 50
      SL(NT3)=1.0E28
      TREAT(NT3)=YM(NTT)
      SS(4)=0.
      SS(3)=0.
      GO TO 51
   50 CONTINUE
      SL(NT3)= SXY(NT3)/SXX(NT3)
      SS(4)= (NT-1.)*SXX(NT3)*SL(NT3)**2
      SS(3)= (SL(NTT)-SL(NT3))**2/(1./(SXX(NT3)*(NT-1.))
     2                            + 1./(SXX(NT2)*(L-NT)))
      TREAT(NT3)=YM(NTT)-SL(NT3)*XM(NTT)
   51 CONTINUE
      SS(1)=SS(1)-SSU
      SS(5)= SS(1)-SS(4)
      SS(2)= (L-1.)*SXX(NTT)*SL(NTT)**2
      SS(10)=SSU+SS(1)+SS(9)
      SS(7) =SSU+SS(1)+SS(6)
      TREAT(NTT)=YM(NTT)-SL(NTT)*XM(NTT)
      SS(11)=SS(12)-SS(10)
      TREAT(NT2)=0
      NDF(1)=NT-1
      NDF(2)= 1
      NDF(3)= 1
      NDF(4)= 1
      NDF(5)= NT-2
      NDF(6)= 1
      NDF(7)= NT+1
      NDF(8)= NT-1
      NDF(9)= NT
      NDF(10)=2*NT
      NDF(12)=L
      NDF(11)=L-NDF(10)
      DO 55 K=1,11
      IF(NDF(K).GT.0) GO TO 54
      XMS(K)=0
      GO TO 55
   54 XMS(K)=SS(K)/NDF(K)
   55 CONTINUE
      DO 60 K=1,10
      F(K)=XMS(K)/XMS(11)
      IF(F(K).GT.0.) GO TO 59
      PROBF(K)=0.
      GO TO 60
   59 PROBF(K)=FISH(F(K),NDF(K),NDF(7))
   60 CONTINUE
      IF(IOPT.LT.1) GO TO 999
      WRITE(6,150)
      WRITE(6,1)
```

```
  1 FORMAT(@0 TREAT@,T10,@NUM@,T18,@INDEPENDENT@,T34,@INDEPENDENT@,
   2       T50,@DEPENDENT@,T67,@DEPENDENT@,T83,@COVARIANCE@,
   3       T99,@INTERCEPT@,T118,@SLOPE@/2X,@MENT@,T10,@BER@,
   4       T20,@MEAN@,T35,@VARIANCE@,T51,@MEAN@,T67,@VARIANCE@//)
    DO 70 I=1,NT
    WRITE(6,2) I,NOB(I),XM(I),SXX(I),YM(I),SYY(I),SXY(I),TREAT(I),
   2           SL(I)
 70 CONTINUE
  2 FORMAT(2X,I5,1X,I5,7(4X,G13.6))
    WRITE(6,4) SXX(NT2),SXY(NT2),SL(NT2)
  4 FORMAT(1X,124(1H-)/2X,@1SLOPE@,26X,G13.6,38X,G13.6,21X,G13.6)
    WRITE(6,5) SXX(NT3),SXY(NT3),SL(NT3)
  5 FORMAT(/2X,@SLOPE @,26X,G13.6,38X,G13.6,21X,G13.6/2X,@OF MEANS@)
    WRITE(6,3)    L     ,XM(NTT),SXX(NTT),YM(NTT),SYY(NTT),SXY(NTT),
   Z              TREAT(NTT),SL(NTT)
  3 FORMAT(/2X,@TOTAL@,1X,T5,7(4X,G13.6))
    WRITE(6,150)
    IF(IOPT.EQ.3) GO TO 80
    WRITE(6,100) SSU,(NDF(I),SS(I),XMS(I),F(I),PROBF(I),I=1, 5)
100 FORMAT(//,15X,93(1H-),/,47X,@ANALYSIS OF COVARIANCE TABLE@,/,15X
   193(1H-),/,15X,@SOURCE OF@,9X,@DEGREES OF@,5X,@SUM OF@,10X,@MEAN@
   212X,@F-@,14X,@PROB@,/,15X,@VARIATION@,9X,@FREEDOM@,8X,@SQUARES@,
   39X,@SQUARE@,10X,@RATIO@,11X,@F@,/,15X,93(1H-),/,15X,@MEAN@,19X,
   *@1@,8X,G13.6,/,15X,@TREATMENTS@,4X,I10,8X,4(G13.6,3X),/,15X,
   493(1H-),/,
   A            15X,@1 POPULATION @,1X,I10,8X,4(G13.6,3X)
   B          ,/,15X,@1 POP.- SLOPE@,1X,I10,8X,4(G13.6,3X)
   C          ,/,15X,@    OF MEANS @,/,15X ,
   * 93(1H-),/,15X,@SLOPE OF MEANS@,I10,8X,4(G13.6,3X)
   D          ,/,15X,@DIFF MEAN SL.@,1X,I10,8X,4(G13.6,3X)
   E          ,/,15X,@AND  PAR. SL.@
   F          )
    WRITE(6,102)      (NDF(I),SS(I),XMS(I),F(I),PROBF(I),I=6,10),
   2             NDF(11),SS(11),XMS(11),NDF(12),SS(12)
102 FORMAT(
   F          ,/,15X,@PARALLEL LINE@,1X,I10,8X,4(G13.6,3X)
   G          ,/,15X,@    SLOPE     @,/, 15X,
   H 93(1H-),/,15X,@PARALLEL LINE@,1X,I10,8X,4(G13.6,3X)
   I          ,/,15X,@    MODEL     @,/,  15X,
   J 93(1H-),/,15X,@NON-             @,1X,I10,8X,4(G13.6,3X)
   K          ,/,15X,@  PARALLELISM@,
   L          ,/,15X,@INDIVIDUAL    @,1X,I10,8X,4(G13.6,3X)
   M          ,/,15X,@  SLOPES      @,/, 15X,
   N 93(1H-),/,15X,@REGRESSION     @,1X,I10,8X,4(G13.6,3X)
   O          ,/,15X,@  MODEL       @
   P          ,/,15X,@ERROR         @,1X,I10,8X,2(G13.6,3X),/,
   * 15X,
   Q 93(1H-),/,15X,@TOTAL          @,1X,I10,8X,  G13.6,/,
   * 15X,
   R 93(1H-))
    WRITE(6,150)
    IF(IOPT.EQ.1) RETURN
 80 NDF(7)=NDF(7)-1
    NDF(10)=NDF(10)-1
    NDF(12)=NDF(12)-1
    SS(7)=SS(7)-SSU
    SS(10)=SS(10)-SSU
    SS(12)=SS(12)-SSU
    XMS(7)=XMS(7)/NDF(7)
    XMS(10)=XMS(10)/NDF(10)
```

```
      F(7)=XMS(7)/XMS(11)
      F(10)=XMS(10)/XMS(11)
      PROBF(7)=FISH(F(7),NDF(7),NDF(11))
      PROBF(10)=FISH(F(10),NDF(10),NDF(11))
      WRITE(6,101)     (NDF(I),SS(I),XMS(I),F(I),PROBF(I),I=1, 5)
  101 FORMAT(//,15X,93(1H-),/,38X,@ANALYSIS OF COVARIANCE TABLE (MEAN DI
     AFFERENCE)@,/,15X,
     193(1H-),/,15X,@SOURCE OF@,9X,@DEGREES OF@,5X,@SUM OF@,10X,@MEAN@,
     212X,@F-@,14X,@PROB@,/,15X,@VARIATION@,9X,@FREEDOM@,8X,@SQUARES@,
     39X,@SQUARE@,10X,@RATIO@,11X,@F@,/,15X,93(1H-),/,
     *                15X,@TREATMENTS@,4X,I10,8X,4(G13.6,3X),/,15X,
     493(1H-),/,
     A             15X,@1 POPULATION @,1X,I10,8X,4(G13.6,3X)
     B           ,/,15X,@1 POP.- SLOPE@,1X,I10,8X,4(G13.6,3X)
     C           ,/,15X,@    OF MEANS @,/,   15X,
     * 93(1H-),/,15X,@SLOPE OF MEANS@,I10,8X,4(G13.6,3X)
     D           ,/,15X,@DIFF MEAN SL.@,1X,I10,8X,4(G13.6,3X)
     E           ,/,15X,@AND  PAR. SL.@
     F           )
      WRITE(6,102)     (NDF(I),SS(I),XMS(I),F(I),PROBF(I),I=6,10),
     2              NDF(11),SS(11),XMS(11),NDF(12),SS(12)
      WRITE(6,150)
  150 FORMAT(1H1)
  999 RETURN
   99 IOPT=-1
      RETURN
      END
-
FOURG
      SUBROUTINE FOURG (DATA,N,ISIGN,WORK)                              FFG
C     COOLEY-TUKEY FAST FOURIER TRANSFORM IN USASI BASIC FORTRAN.       FFG
C     ONE-DIMENSIONAL TRANSFORM OF COMPLEX DATA, ARBITRARY NUMBER OF    FFG
C     POINTS.  N POINTS CAN BE TRANSFORMED IN TIME PROPORTIONAL TO      FFG
C     N*LOG(N) (FOR N NON-PRIME), WHEREAS OTHER METHODS TAKE N**2 TIME. FFG
C     FURTHERMORE, BECAUSE FEWER ARITHMETIC OPERATIONS ARE PERFORMED,   FFG
C     LESS ERROR IS BUILT UP.  THE TRANSFORM DONE IS--                  FFG
C     DIMENSION DATA(N),TRANSFORM(N),WORK(N)                            FFG
C     COMPLEX DATA,TRANSFORM,WORK                                       FFG
C     TRANSFORM(K) = SUM(DATA(J)*EXP(ISIGN*2*PI*I*(J-1)*(K-1)/N)),      FFG
C     SUMMED FROM J = 1 TO N FOR ALL K FROM 1 TO N.  THE TRANSFORM      FFG
C     VALUES ARE RETURNED TO DATA, REPLACING THE INPUT.  N MAY BE ANY   FFG
C     POSITIVE NUMBER, BUT IT SHOULD BE NON-PRIME FOR SPEED.  ISIGN =   FFG
C     +1 OR -1.  A -1 TRANSFORM FOLLOWED BY A +1 ONE (OR VICE VERSA)    FFG
C     RETURNS N TIMES THE ORIGINAL DATA.  WORK IS A ONE-DIMENSIONAL     FFG
C     COMPLEX ARRAY OF LENGTH N USED FOR WORKING STORAGE.               FFG
C     RUNNING TIME IS PROPORTIONAL TO N * (SUM OF THE PRIME FACTORS OF  FFG
C     N).  FOR EXAMPLE, N = 1960, TIME IS TO * 1960 * (2+2+2+5+7+7).    FFG
C     NAIVE METHODS DIRECTLY IMPLEMENTING THE SUMMATION RUN IN TIME     FFG
C     PROPORTIONAL TO N**2.  AN UPPER BOUND FOR THE RMS RELATIVE ERROR  FFG
C     IS 3 * 2**(-B) * SUM(F**1.5), WHERE B IS THE NUMBER OF BITS IN    FFG
C     THE FLOATING POINT FRACTION AND THE SUM IS OVER THE PRIME         FFG
C     FACTORS OF N.  WRITTEN BY NORMAN BRENNER, MIT LINCOLN LABORATORY, FFG
C     AUGUST 1968.  SEE--IEEE TRANSACTIONS ON AUDIO AND ELECTROACOUSTICSFFG
C     (JUNE 1967), SPECIAL ISSUE ON THE FAST FOURIER TRANSFORM.         FFG
      DIMENSION DATA(1), WORK(1), IFACT(32)                             FFG
      TWOPI=6.283185307*FLOAT(ISIGN)                                    FFG
C     FACTOR N INTO ITS PRIME FACTORS, NFACT IN NUMBER.  FOR EXAMPLE,   FFG
C     FOR N = 1960, NFACT = 6 AND IFACT(IF) = 2, 2, 2, 5, 7 AND 7.      FFG
      IF=0                                                              FFG
      NPART=N                                                           FFG
```

```
      DO 50 ID=1,N,2                                                    FFG  32
      IDIV=ID                                                           FFG  33
      IF (ID-1) 10,10,20                                                FFG  34
 10   IDIV=2                                                            FFG  35
 20   IQUOT=NPART/IDIV                                                  FFG  36
      IF (NPART-IDIV*IQUOT) 40,30,40                                    FFG  37
 30   IF=IF+1                                                           FFG  38
      IFACT(IF)=IDIV                                                    FFG  39
      NPART=IQUOT                                                       FFG  40
      GO TO 20                                                          FFG  41
 40   IF (IQUOT-IDIV) 60,60,50                                          FFG  42
 50   CONTINUE                                                          FFG  43
 60   IF (NPART-1) 80,80,70                                             FFG  44
 70   IF=IF+1                                                           FFG  45
      IFACT(IF)=NPART                                                   FFG  46
 80   NFACT=IF                                                          FFG  47
C     SHUFFLE THE DATA ARRAY BY REVERSING THE DIGITS OF THE INDEX.      FFG  48
C     REPLACE DATA(I) BY DATA(IREV) FOR ALL I FROM 1 TO N.  IREV-1 IS   FFG  49
C     THE INTEGER WHOSE DIGIT REPRESENTATION IN THE MULTI-RADIX         FFG  50
C     NOTATION OF FACTORS IFACT(IF) IS THE REVERSE OF THE REPRESENTATIONFFG  51
C     OF I-1.  FOR EXAMPLE, IF ALL IFACT(IF) = 2, THEN FOR I-1 = 11001, FFG  52
C     IREV-1 = 10011.  A WORK ARRAY OF LENGTH N IS NEEDED.              FFG  53
      IP0=2                                                             FFG  54
      IP3=IP0*N                                                         FFG  55
      IWORK=1                                                           FFG  56
      I3REV=1                                                           FFG  57
      DO 110 I3=1,IP3,IP0                                               FFG  58
      WORK(IWORK)=DATA(I3REV)                                           FFG  59
      WORK(IWORK+1)=DATA(I3REV+1)                                       FFG  60
      IP2=IP3                                                           FFG   1
      DO 100 IF=1,NFACT                                                 FFG   2
      IP1=IP2/IFACT(IF)                                                 FFG  62
      I3REV=I3REV+IP1                                                   FFG  64
      IF (I3REV-IP2) 110,110,90                                         FFG  65
 90   I3REV=I3REV-IP2                                                   FFG  66
100   IP2=IP1                                                           FFG  67
110   IWORK=IWORK+IP0                                                   FFG  68
      IWORK=1                                                           FFG  69
      DO 120 I3=1,IP3,IP0                                               FFG  70
      DATA(I3)=WORK(IWORK)                                              FFG  71
      DATA(I3+1)=WORK(IWORK+1)                                          FFG  72
120   IWORK=IWORK+IP0                                                   FFG  73
C     PHASE-SHIFTED FOURIER TRANSFORM OF LENGTH IFACT(IF).              FFG  74
C     IPROD=IP1/IP0                                                     FFG  75
C     IREM=N/(IFACT(IF)*IPROD)                                          FFG  76
C     DIMENSION DATA(IPROD,IFACT(IF),IREM),WORK(IFACT(IF))              FFG  77
C     COMPLEX DATA,WORK                                                 FFG  78
C     DATA(I1,J2,I3) = SUM(DATA(I1,I2,I3) * W**(I2-1)), SUMMED OVER     FFG  79
C     I2 = 1 TO IFACT(IF) FOR ALL I1 FROM 1 TO IPROD, J2 FROM 1 TO      FFG  80
C     IFACT(IF) AND I3 FROM 1 TO IREM.                                  FFG  81
C     W = EXP(ISIGN*2*PI*I*(I1-1+IPROD*(J2-1))/(IPROD*IFACT(IF))).      FFG  82
      IF=0                                                              FFG  83
      IP1=IP0                                                           FFG  84
130   IF (IP1-IP3) 140,240,240                                          FFG  85
140   IF=IF+1                                                           FFG  86
      IFCUR=IFACT(IF)                                                   FFG
      IP2=IP1*IFCUR                                                     FFG
      THETA=TWOPI/FLOAT(IFCUR)                                          FFG  89
      SINTH=SIN(THETA/2.)                                               FFG  90
      ROOTR=-2.*SINTH*SINTH                                             FFG  91
```

```
C       COS(THETA)-1, FOR ACCURACY                                      FFG  92
        ROOTI=SIN(THETA)                                                FFG  93
        THETA=TWOPI/FLOAT(IP2/IP0)                                      FFG  94
        SINTH=SIN(THETA/2.)                                             FFG  95
        WSTPR=-2.*SINTH*SINTH                                           FFG  96
        WSTPI=SIN(THETA)                                                FFG  97
        WMINR=1.                                                        FFG  98
        WMINI=0.                                                        FFG  99
        DO 230 I1=1,IP1,IP0                                             FFG 100
        IF (IFCUR-2) 150,150,170                                        FFG 101
  150   DO 160 I3=I1,IP3,IP2                                            FFG 102
        J0=I3                                                           FFG 103
        J1=I3+IP1                                                       FFG 104
        TEMPR=WMINR*DATA(J1)-WMINI*DATA(J1+1)                           FFG 105
        TEMPI=WMINR*DATA(J1+1)+WMINI*DATA(J1)                           FFG 106
        DATA(J1)=DATA(J0)-TEMPR                                         FFG 107
        DATA(J1+1)=DATA(J0+1)-TEMPI                                     FFG 108
        DATA(J0)=DATA(J0)+TEMPR                                         FFG 109
  160   DATA(J0+1)=DATA(J0+1)+TEMPI                                     FFG 110
        GO TO 220                                                       FFG 111
  170   IWMAX=IP0*IFCUR                                                 FFG 112
        DO 210 I3=I1,IP3,IP2                                            FFG 113
        I2MAX=I3+IP2-IP1                                                FFG 114
        WR=WMINR                                                        FFG 115
        WI=WMINI                                                        FFG 116
        DO 200 IWORK=1,IWMAX,IP0                                        FFG 117
        I2=I2MAX                                                        FFG 118
        SUMR=DATA(I2)                                                   FFG 119
        SUMI=DATA(I2+1)                                                 FFG 120
  180   I2=I2-IP1                                                       FFG 121
        TEMPR=SUMR                                                      FFG 122
        SUMR=WR*SUMR-WI*SUMI+DATA(I2)                                   FFG 123
        SUMI=WR*SUMI+WI*TEMPR+DATA(I2+1)                                FFG 124
        IF (I2-I3) 190,190,180                                          FFG 125
  190   WORK(IWORK)=SUMR                                                FFG 126
        WORK(IWORK+1)=SUMI                                              FFG 127
        TEMPR=WR                                                        FFG 128
        WR=WR*ROOTR-WI*ROOTI+WR                                         FFG 129
  200   WI=TEMPR*ROOTI+WI*ROOTR+WI                                      FFG 130
        IWORK=1                                                         FFG 131
        DO 210 I2=I3,I2MAX,IP1                                          FFG 132
        DATA(I2)=WORK(IWORK)                                            FFG 133
        DATA(I2+1)=WORK(IWORK+1)                                        FFG 134
  210   IWORK=IWORK+IP0                                                 FFG 135
  220   TEMPR=WMINR                                                     FFG 136
        WMINR=WMINR*WSTPR-WMINI*WSTPI+WMINR                             FFG 137
  230   WMINI=TEMPR*WSTPI+WMINI*WSTPR+WMINI                             FFG 138
        IP1=IP2                                                         FFG 139
        GO TO 130                                                       FFG 140
  240   RETURN                                                          FFG 141
        END                                                             FFG 142
-
LAGCOR
        SUBROUTINE LAGCOR(LA,A,LB,B,C,LSTART,LSTOP)
C
C       THIS ROUTINE CALCULATES A SAMPLE CROSS-CORRELATION OF THE RECORD
C            A OVER THE RECORD B WITH LAGS BETWEEN LSTART AND LSTOP AND
C            STORES THE RESULT IN C
C     **** CAUTION ***** THERE IS NO CHECK FOR A ZERO RECORD
C
```

```
      DIMENSION A(LA),B(LB),C(LA)
      DO 50 J=LSTART,LSTOP
      U=0.0
      SUMA=0.0
      SUMB=0.0
      SA=0.0
      SB=0.0
      IF(LB-(LA-J+1)) 10,10,20
   10 N=LB
      GO TO 30
   20 N=LA-J+1
      IF(N.GT.0) GO TO 30
      DO 25 I=J,LSTOP
   25 C(I)=-2.
      RETURN
   30 EN=N
      DO 40 I=1,N
      IJ=I+J-1
      SUMA=SUMA+A(IJ)
      SUMB=SUMB+B(I)
      SA=SA+A(IJ)*A(IJ)
      SB=SB+B(I)*B(I)
   40 U=U+A(IJ)*B(I)
      SUMA=SUMA/EN
      SUMB=SUMB/EN
      SA=SA-SUMA*SUMA*EN
      SB=SB-SUMB*SUMB*EN
   50 C(J)=(U-EN*SUMA*SUMB)/SQRT(SA*SB)
      RETURN
      END
-
-                    END-OF-RECORD
-                    END-OF-INFORMATION
```

D.9 Program RETD as modified to perform the basic analysis plus and analysis of covariance

```
      PROGRAM RETD(INPUT,OUTPUT,TAPE6=OUTPUT)                          FIX
C     MAIN PROGRAM                                                     RETD
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP       XRED
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)         RETD
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)          XREDUC
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),  RETD
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA          RETD
      COMMON/RG/ DATESC(3,5),IVV(4), SSMEAN(3), SSIGMA(3), NNN(3),SSE(3)RETD
      COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,     RETD
     +                MFUNC,IFILE,IERR                                 RETD
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)          RETD
C     START,LAUNCH,SPLASH DOWN                                         JAN25
      DATA DATESC/180.,208.,265.,114.,144.,172.,189.,209.,268.,        DEC10
     X 283.,320.,404.,  290.,342.,354./                               DEC10
C     SL4 SPLASH DOWN IS 39  FEB 8 1974                                DEC10
      IOPLOT = 0                                                       RETD
      KSKIP = 0                                                        RETD
C     IFFF IS LOOP CONTROL TO ALLOW OVERLAY                            JAN25
      IFFF = -1                                                        RETD
    1 CALL RET                                                         RETD
      CALL STAT                                                        RETD
      GO TO 1                                                          RETD
      END                                                              RETD
      SUBROUTINE LINE(X,Y,N,M,L,LL)                                    LINE
C     X IS THE DATE. (INDEPENDENT VARIABLE)                            LINE
C     Y IS THE DATA. (DEPENDENT VARIABLE)                              LINE
C     IFLAG = 13   LAST POINT NOT VALID                                LINE
C     JFLAG = 13   LAST POINT VALID                                    LINE
C     ISFLAG IS THE NUMBER OF POINTS PLOTED.                           LINE
      DIMENSION X(1), Y(1)                                             LINE
      DATA D/0.04/                                                     LINE
      IFLAG = 0                                                        LINE
      JFLAG = 0                                                        LINE
      ISFLAG = 0                                                       LINE
      IP = 3                                                           LINE
      IF(N - 1) 9,1,1                                                  LINE
    1 DO 6 I=1,N                                                       LINE
      II = I*M - M + 1                                                 LINE
      XP = (X(II) - X(N + M))/X(N + 2*M)                               LINE
      YP = (Y(II) - Y(N + M))/Y(N + 2*M)                               LINE
C     CHECK TO SEE IF POINT VALUE IS VALID.                            LINE
      IF(Y(II)) 3,13,23                                                LINE
    3 IY = -Y(II) + 0.5                                                LINE
      IF(IY - 2) 13,14,13                                              LINE
C     THIS IS ZERO ON THE GRAPH.                                       LINE
   14 YP = -Y(N+1)/Y(N+2*M)                                            LINE
      GO TO 23                                                         LINE
C     INVALID POINT.                                                   LINE
   13 IFLAG = 13                                                       LINE
      GO TO 6                                                          LINE
C     VALID POINT.  CHECK FOR CONTINUOUS DATE.                         LINE
   23 JFLAG = 13                                                       LINE
      IF(ISFLAG) 25,25,24                                              LINE
   24 IF(X(II) - X(II-M) - 1.5) 25,35,35                               LINE
   25 IF(IFLAG) 35,26,35                                               LIN
C     PLOT THIS PEN DOWN.                                              LIN
   26 CALL PLOT(XP,YP,IP)                                              LINE
      ISFLAG = ISFLAG + 1                                              LINE
      IP = 2                                                           LINE
      GO TO 6                                                          LINE
```

```
   DISCONTINUOUS GRAPH, PLOT AN X ON EACH END OF LINE.                    LINE
35 CALL WHERE(U,V,F)                                                      LINE
   IFLAG = 0                                                              LINE
   KFLAG = JFLAG*ISFLAG                                                   LINE
   IF(KFLAG) 16,17,16                                                     LINE
16  CALL SYMBOL(U,V,D,LL,0.0,-1)                                          LINE
   ISFLAG = ISFLAG + 1                                                    LINE
17 CALL SYMBOL(XP,YP,D,LL,0.0,-1)                                         LINE
   IF = 2                                                                 LINE
   ISFLAG = ISFLAG + 1                                                    LINE
 6 CONTINUE                                                               LINE
   KFLAG = IFLAG*JFLAG                                                    LINE
   IF(KFLAG) 9,9,29                                                       LINE
29 CALL WHERE(U,V,F)                                                      LINE
   CALL SYMBOL(U,V,D,LL,0.0,-1)                                           LINE
   ISFLAG = ISFLAG + 1                                                    LINE
 9 RETURN                                                                 LINE
   END                                                                    LINE
   SUBROUTINE AXIS (X,Y,BCD,NC,SIZE,THETA,YMIN,DY,NDEC,NLAB,NTIC)         AXIS
   DIMENSION G(2), H(11)                                                  AXIS
   DATA G/.8,.56/                                                         AXIS
   DATA H/.56,.4,.28,.2,.14,.1,.07,.05,.035,.025,.0175/                   AXIS
   AC = NC                                                                AXIS
   SIG=SIGN(1.0,AC)                                                       AXIS
 2 NAC=IABS(NC)                                                           AXIS
   TH=THETA*0.017453294                                                   AXIS
   IF(NLAB.LE.0) NLAB = 1                                                 AXIS
   IF(NTIC.LE.0) NTIC = 1                                                 AXIS
   FNLAB = NLAB                                                           AXIS
   N = SIZE*FNLAB + 0.1                                                   AXIS
   N = SIZE + 0.50                                                        AXIS
   CTH = COS  (TH)                                                        AXIS
   STH = SIN  (TH)                                                        AXIS
   CTN = CTH/FNLAB                                                        AXIS
   STN = STH/FNLAB                                                        AXIS
   TN = N                                                                 AXIS
   N1 = N + 1                                                             AXIS
   N2 = N1/2                                                              AXIS
   ADY=ABS(DY/FNLAB)                                                      AXIS
   ADY=ABS(DY)                                                            AXIS
   STAT=YMIN                                                              AXIS
   EXP = 0.0                                                              AXIS
   IF ( ADY ) 9,18,9                                                      AXIS
 9 IF ( ADY -100.0 ) 10,12,12                                             AXIS
12 ADY = ADY / 10.0                                                       AXIS
   STAT=STAT/10.0                                                         AXIS
   EXP = EXP + 1.0                                                        AXIS
   GO TO 9                                                                AXIS
14 ADY = ADY * 10.0                                                       AXIS
   STAT=STAT*10.0                                                         AXIS
   EXP = EXP - 1.0                                                        AXIS
10 IF ( ADY - 1.00 ) 14,18,18                                             AXIS
10 IF ( ADY - 0.01 ) 14,18,18                                             AXIS
18 XA = X  - (.20 * SIG  - .05) *STH - .0857 * CTH                        AXIS
18 XA = X  - (H(NLAB+1)  * SIG  - H(NLAB+5)) *STH - .0857 * CTH           AXIS
   YA = Y  + (.20 * SIG  - .05) * CTH - .0857 * STH                       AXIS
   YA = Y  + (H(NLAB+1) * SIG  - H(NLAB+5)) * CTH - .0857 * STH           AXIS
   I = 0                                                                  AXIS
25 I = I + 1                                                              AXIS
   CALL NUMBER (XA,YA,0.1,STAT,THETA,2)                                   AXIS
```

```
      CALL NUMBER (XA,YA,H(NLAB+3),STAT,THETA,NDEC)                    AXIS
      STAT=STAT+SIGN(ADY,DY)                                           AXIS
C     XA = X  + CTH                                                    AXIS
      XA = X  + CTN                                                    AXIS
C     YA = YA + STH                                                    AXIS
      YA = YA + STN                                                    AXIS
      IF(I - N2) 25,31,26                                              AXIS
   26 IF(I - N1) 25,60,60                                              AXIS
   31 TNC = NAC + 7                                                    AXIS
C     XC = X + (SIZE / 2.0 -.06 * TNC)*CTH - (-.07 + SIG *.36)* STH    AXIS
      XC = X + (SIZE / 2.0 -H(NLAB+4) * TNC)*CTH                       AXIS
     1- (-H(NLAB+4) + SIG *(H(NLAB) + H(NLAB+3)))* STH                 AXIS
C     YC = Y + (SIZE / 2.0 -.06 * TNC)*STH + (-.07 + SIG *.36)* CTH    AXIS
      YC = Y + (SIZE / 2.0 -H(NLAB+4) * TNC)*STH                       AXIS
     1+ (-H(NLAB+4) + SIG *(H(NLAB) + H(NLAB+3)))* CTH                 AXIS
C     CALL SYMBOL (XC,YC,0.14,BCD,THETA,NAC)                           AXIS
      CALL SYMBOL (XC,YC,H(NLAB+2),BCD,THETA,NAC)                      AXIS
      XC = XC +((TNC -6.0) * 0.12)* CTH                                AXIS
      YC = YC +((TNC -6.0) * 0.12)* STH                                AXIS
      IF (EXP) 35,50,35                                                AXIS
C  35 CALL SYMBOL (XC,YC,0.14,@(X10  )@ ,THETA,7)                      AXIS
C  35 CALL SYMBOL (XC,YC,H(NLAB+2),@(X10  )@ ,THETA,7)                 AXIS
   35 CALL SYMBOL (XC,YC,H(NLAB+2),7H(X10  ),THETA,7)                  AXIS
C     XC = XC + .48 * CTH  -.07 * STH                                  AXIS
      XC=XC+.38*CTH-H(NLAB+4)*STH                                      DEC1
C     YC = YC + .48 * STH  +.07 * CTH                                  AXIS
      YC=YC+.38*STH+H(NLAB+4)*CTH                                      DEC1
C  40 CALL NUMBER (XC,YC,0.10,EXP,THETA,-1)                            AXIS
   40 CALL NUMBER (XC,YC,H(NLAB+3),EXP,THETA,-1)                       AXIS
   50 GO TO 25                                                         AXIS
   60 FNTIC = NTIC                                                     AXIS
      NT = N*NTIC                                                      AXIS
      TN =.NT                                                          AXIS
      CTH = CTN/FNTIC                                                  AXIS
      STH = STN/FNTIC                                                  AXIS
      XB = X + TN*CTH                                                  AXIS
      YB = Y + TN*STH                                                  AXIS
      XDELT = - H(6 - NTIC) * SIG  * STH                               AXIS
      YDELT = H(6 - NTIC) * SIG  * CTH                                 AXIS
      XA = XB + XDELT + XDELT                                          AXIS
C     XA = XB- 0.1 * SIG  * STH                                        AXIS
      YA = YB + YDELT + YDELT                                          AXIS
C     YA = YB+ 0.1 * SIG  * CTH                                        AXIS
      CALL PLOT (XA,YA,3)                                              AXIS
      XA = XA - XDELT                                                  AXIS
      YA = YA - YDELT                                                  AXIS
C     DO 20  I =1,N                                                    AXIS
      DO 20  I =1,N                                                    AXIS
      DO 20 II =1,NTIC                                                 AXIS
      IF(II.LT.NTIC) GO TO 45                                          AXIS
      XX = XDELT                                                       AXIS
      YY = YDELT                                                       AXIS
      GO TO 46                                                         AXIS
   45 XX = 0.                                                          AXIS
      YY = 0.                                                          AXIS
   46 CONTINUE                                                         AXIS
      CALL PLOT (XB,YB,2)                                              AXIS
      XC = XB - CTH                                                    AXIS
      YC = YB - STH                                                    AXIS
      CALL PLOT (XC,YC,2)                                              AXIS
```

```
      XA = XA - CTH                                                    AXIS
      XA = XA - CTH + XX                                               AXIS
      YA = YA - STH                                                    AXIS
      YA = YA - STH + YY                                               AXIS
      CALL PLOT (XA,YA,2)                                              AXIS
      XA = XA - XX                                                     AXIS
      YA = YA - YY                                                     AXIS
      XB = XC                                                          AXIS
   20 YB = YC                                                          AXIS
      RETURN                                                           AXIS
      END                                                              AXIS
      SUBROUTINE SCALE (X,S,N,K)                                       SCALE
C                                                                      SCALE
C  WHERE-  X  IS THE NAME OF THE ARRAY OF DATA TO BE SCANNED FOR MAXIMUMSCALE
C             AND MINIMUM VALUES. AN ADJUSTED MINIMUM VALUE WILL BE    SCALE
C             STORED IN X(N*K+1). AN ADJUSTED DX(MAX.-MIN.) WILL BE    SCALE
C             STORED IN X(N*K+K+1).                                    SCALE
C          S  IS THE LENGTH OVER WHICH THIS DATA IS TO BE PLOTTED.     SCALE
C          N  IS THE NUMBER OF DATA POINTS IN THE ARRAY X.             SCALE
C          K  IS THE REPEAT CYCLE OF A MIXED ARRAY.(NORMALLY 1)        SCALE
C                                                                      SCALE
      DIMENSION X(2)                                                   SCALE
      IT = 13                                                          SCALE
      NP = N * K                                                       SCALE
      L = NP + 1                                                       SCALE
      J = NP +K + 1                                                    SCALE
      XMAX = X(1)                                                      SCALE
      X(L) = X(1)                                                      SCALE
      DO 2 I=1,NP,K                                                    SCALE
      IF(X(I)) 2,2,3                                                   SCALE
    2 CONTINUE                                                         SCALE
    3 XMAX = X(I)                                                      SCALE
      X(L)= X(I)                                                       SCALE
      X(L)= 0.0                                                        SCALE
      DO 10 I = 1,NP,K                                                 SCALE
      IF(X(I)) 22,10,23                                                SCALE
   22 ITEST = 2.5 - X(I)                                               SCALE
      IF(ITEST) 10,24,10                                               SCALE
   24 X(L) = 0.0                                                       SCALE
      IT = 0                                                           SCALE
      GO TO 10                                                         SCALE
   23 IF (XMAX-X(I)) 5,6,6                                             SCALE
    5 XMAX = X(I)                                                      SCALE
    6 IF (X(L)-X(I))10,10,7                                            SCALE
    7 X(L) = X(I)                                                      SCALE
   10 CONTINUE                                                         SCALE
      DX = (XMAX - X(L)) /S                                            SCALE
      IF (DX) 31,31,30                                                 SCALE
   31 X(J) = 1.0                                                       SCALE
      X(L) = X(L) - 0.5                                                SCALE
      RETURN                                                           SCALE
   30 IDX = ALOG10 (DX)                                                SCALE
      IXMN = X(L) * 10.0 ** (-IDX)                                     SCALE
      IF (X(L))32,33,34                                                SCALE
   32 IXMN = X(L) * 10.0 ** (-IDX) - 0.99                              SCALE
   34 X(L) = IXMN                                                      SCALE
      X(L) = X(L) * 10.0 ** IDX                                        SCALE
   33 DX = ALOG10 ((XMAX-X(I))/S)                                      SCALE
      IDX = DX                                                         SCALE
      XMAX = IDX                                                       SCALE
```

```
      DX=10.0**(DX-XMAX)                                                SCALE
      XMAX = 1.0                                                        SCALE
   41 IF (DX-1.0) 40,20,11                                              SCAL
   40 DX = DX * 10.0                                                    SCAL
      IDX = IDX - 1                                                     SCALE
      GO TO 41                                                          SCALE
   11 XMAX=2.0                                                          SCALE
      IF(DX-2.0) 20,20,12                                               SCALE
   12 XMAX = 4.0                                                        SCALE
      IF (DX-4.0) 20,20,13                                              SCALE
   13 XMAX=5.0                                                          SCALE
      IF(DX-5.0) 20,20,14                                               SCALE
   14 XMAX=8.0                                                          SCALE
      IF (DX-8.0) 20,20,15                                              SCALE
   15 XMAX=10.0                                                         SCALE
   20 X(J) = XMAX * 10.0 ** IDX                                         SCALE
      IF(IT) 49,49,39                                                   SCALE
   39 CONTINUE                                                          SCALE
      IMAX = XMAX + 0.00001                                             SCALE
      IXMN = X(L) * 10.0 ** (-IDX)                                      SCALE
      IMOD = IXMN/IMAX*IMAX                                             SCALE
      X(L) = IMOD                                                       SCALE
      X(L) = X(L) * 10.0 ** IDX                                         SCALE
      GO TO 59                                                          SCALE
   49 CONTINUE                                                          SCALE
      X(L) = 0.0                                                        SCALE
   59 CONTINUE                                                          SCALE
      RETURN                                                            SCALE
      END                                                               SCAL
      SUBROUTINE RET                                                    RET
C                                                                       RET
C MSC ENDOCRINE DATA RETREIVAL PROGRAM                                  RET
C     FIRST LOOK STATISTICS PROGRAM.                                    RET
C                                                                       RET
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP        XREDUC
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)          RET
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)           XREDUC
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),   RET
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA           RET
C     COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,      RET
      COMMON /SISBUF/ JBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,      RET
     +                MFUNC,IFILE,IERR                                  RET
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)           RET
      DIMENSION IBUF(14)                                                RET
      EQUIVALENCE(IBUF,T)                                               RET
      EXTERNAL ERROR                                                    RET
C                                                                       RET
C RA1=SAMPLE DIRECTORY  ***  RA2=URINE FILE  ***  RA3=BLOOD FILE        RET
C                       ***  RA4=DAILY FILE  ***                       RET
C                                                                       RET
C                                                                       RET
C PLACE KEYED FILES IN UPDATE MODE EXCLUDING @RA1@ WHICH IS             RET
C             TO BE USED AS A SEQUENTIAL SEARCH FILE                    RET
C             ASSIGNED TO FORTRAN UNIT NO. 1                            RET
C                                                                       RET
      IF(IFFF) 31,201,140                                               RET
C  31 CALL RANDOM(4,@RA2@,125,5,7,100 ,100 ,ICODE,LOC)                  RET
   31 CONTINUE                                                          RET
      CALL FILEIS(FITRA1,3LLFN,3LRA1,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL, 350,3LMNR, 350,3LWSA, T)                       RET
```

```
      CALL STOREF(FITRA1,3LERL,20)                                      RET
      CALL STOREF(FITRA1,2LEX,  ERROR)                                  RET
      CALL OPENM(FITRA1,3LI-O)                                          RET
      CALL REWND(FITRA1)                                                RET
      CALL FILEIS(FITRA2,3LLFN,3LRA2,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL,1250,3LMNR,1250,3LWSA, T)                       RET
      CALL STOREF(FITRA2,3LERL,0)                                       RET
      CALL STOREF(FITRA2,2LEX,  ERROR)                                  RET
      CALL OPENM(FITRA2,5LINPUT)                                        RET
      CALL REWND(FITRA2)                                                RET
      CALL FILEIS(FITRA3,3LLFN,3LRA3,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL, 350,3LMNR, 350,3LWSA, T)                       RET
      CALL STOREF(FITRA3,3LERL,0)                                       RET
      CALL STOREF(FITRA3,2LEX,  ERROR)                                  RET
      CALL OPENM(FITRA3,5LINPUT)                                        RET
      CALL REWND(FITRA3)                                                RET
      CALL FILEIS(FITRA4,3LLFN,3LRA4,2LKA,KEY,2LKL,10,2LKT,1LI,2LRB,5   RET
     +           ,3LMRL, 300,3LMNR, 300,3LWSA,T)                        RET
      CALL STOREF(FITRA4,3LERL,0)                                       RET
      CALL STOREF(FITRA4,2LEX,  ERROR)                                  RET
      CALL OPENM(FITRA4,5LINPUT)                                        RET
      CALL REWND(FITRA4)                                                RET
      IFFF = 0                                                          RET
C                                                                       RET
C INITIALIZE COUNTER FLAGS FOR THE FIRST PASS OF                        RET
C    ANY GIVEN RETREIVAL                                                RET
C                                                                       RET
  201 ISFLG = 1                                                         RET
      IDFLG = 1                                                         RET
      IMFLG = 1                                                         RET
      ITPFLG = 1                                                        RET
      ITTFLG = 1                                                        RET
      IFFF = 1                                                          RET
C                                                                       RET
C READ ONE DATA CARD                                                    RET
C                                                                       RET
    1 READ 200,(ISAV(I),I=1,16)                                         RET
  200 FORMAT(A4,1X,A4,1X,14I5)                                          RET
C                                                                       RET
C DETERMINE CARD TYPE AND COUNTER SEQUENCE                              RET
C                                                                       RET
      IF(ISAV(1).EQ.4HJOB ) GO TO 19                                    RET
      IF(ISAV(1).EQ.4HSTUD.AND.ISFLG.EQ.1) GO TO  5                     RET
      IF(ISAV(1).EQ.4HSTUD.AND.ISFLG.EQ.0) GO TO 45                     RET
      IF(ISAV(1).EQ.4HDATE.AND.IDFLG.EQ.1) GO TO 6                      RET
      IF(ISAV(1).EQ.4HDATE.AND.IDFLG.EQ.0) GO TO 46                     RET
      IF(ISAV(1).EQ.4HMAN .AND.IMFLG.EQ.1) GO TO  7                     RET
      IF(ISAV(1).EQ.4HMAN .AND.IMFLG.EQ.0) GO TO 47                     RET
      IF(ISAV(1).EQ.4HTYPE.AND.ITPFLG.EQ.1) GO TO  8                    RET
      IF(ISAV(1).EQ.4HTYPE.AND.ITPFLG.EQ.0) GO TO 48                    RET
      IF(ISAV(1).EQ.4HTEST.AND.ITTFLG.EQ.1) GO TO  9                    RET
      IF(ISAV(1).EQ.4HTEST.AND.ITTFLG.EQ.0) GO TO 49                    RET
      IF(ISAV(1).EQ.4HEND ) GO TO 40                                    RET
      IF(ISAV(1).EQ.4HEOF ) GO TO 99                                    RET
C                                                                       RET
C BUILD RETREIVAL CRITERIA ARRAYS                                       RET
C                                                                       RET
    5 JSTALL=0                                                          RET
      IS = 0                                                            RET
      ISFLG = 0                                                         RET
```

```
   45 CALL SETCRI(LSTUDY,IS,JSTALL)                                    RET
C     OUTPUT,(LSTUDY(I),I=1,5),IS,JSTALL                               RET
      GO TO 1                                                          RET
    6 JDTALL=0                                                         RET
      ID = 0                                                           RET
      IDFLG = 0                                                        RET
   46 CALL SETCRI(LDATE,ID,JDTALL)                                     RET
C     OUTPUT,(LDATE(I),I=1,10),ID,JDTALL                               RET
      GO TO 1                                                          RET
    7 MANALL=0                                                         RET
      IM = 0                                                           RET
      IMFLG = 0                                                        RET
   47 CALL SETCRI(LMAN,IM,MANALL)                                      RET
C     OUTPUT,(LMAN(I),I=1,9),IM,MANALL                                 RET
      GO TO 1                                                          RET
    8 JTPALL=0                                                         RET
      ITP = 0                                                          RET
      ITPFLG = 0                                                       RET
   48 CALL SETCRI(LTYPE,ITP,JTPALL)                                    RET
C     OUTPUT,(LTYPE(I),I=1,10),ITP,JTPALL                              RET
      GO TO 1                                                          RET
    9 JTTALL=0                                                         RET
      ITT = 0                                                          RET
      ITTFLG = 0                                                       RET
   49 CALL SETCRI(LTEST,ITT,JTTALL)                                    RET
      ITT=1                                                            RET
C     IF(JTTALL.EQ.1) ITT = 124                                        RET
      LSTART = ISAV(3)                                                 RET
      LSTOP  = ISAV(4)                                                 RET
      IF(LSTOP.EQ.0) LSTOP=LSTART                                      DEC1
      GO TO 1                                                          RET
   19 DO 119 I=1,16                                                    RET
  119 JOB(I)=ISAV(I)                                                   RET
      IF(IOPLOT.EQ.0.AND.JOB(3).GE.1000) CALL PLOTS(14HNASA ENDOCRINE,  RET
     1  14)                                                            RET
      IF(JOB(3).GE.1000) IOPLOT = 999                                  RET
      IF(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH.OR.JOB(2).EQ.4HVOLU)GOTO1 DEC10
      JOB(2)=4HDATE                                                    DEC10
      PRINT 601                                                        DEC10
  601 FORMAT(* UNKNOWN JOB PARAMETER DATE ASSUMED*)                    DEC10
      GO TO 1                                                          RET
C                                                                      RET
C BUILD HIT ARRAY OF SAMPLE NUMBERS                                    RET
C                                                                      RET
   40 CONTINUE                                                         RET
      LT = LSTART - 1                                                  RET
C                                                                      RET
C     DO 4321 LT=LSTART,LSTOP                                          RET
  140 IF(IFFF - 2) 145,240,145                                         RET
  145 LT = LT + 1                                                      RET
C                                                                      RET
      IF(LT - LSTOP) 150,150,4321                                      RET
  150 LM = 0                                                           RET
      IFFF = 2                                                         RET
C     IF(KOT.EQ.0) CALL SSWTCH(1,KKT)                                  RET
C     IF(KKT.EQ.1) GO TO 999                                           RET
      LTEST(1) = LT                                                    RET
C     DO 4321 LM=1,3                                                   RET
  240 LM = LM + 1                                                      RET
      IF(LM - 3) 250,250,245                                           RET
```

```
  245 IFFF = 1                                                          RET
      GO TO 140                                                         RET
  250 LMAN(1) = LM                                                      RET
      IH = 1                                                            RET
C     REWIND 1                                                          RET
      CALL REWND(FITRA1)                                                RET
C                                                                       RET
C                                                                       RET
C THE KEYED (RA1) SAMPLE DIRECTORY FILE IS TREATED AS A SEQUENTIAL      RET
C     FILE DURING THE SEARCH FOR SAMPLES WHICH SATISFY                  RET
C                     THE INPUT CRITERIA                                RET
C                                                                       RET
C   10 BUFFER IN(1,1)(IBUF(1),IBUF(14))                                 RET
   10 CALL GETN(FITRA1)                                                 RET
      IF(IFETCH(FITRA1,2LFP).EQ.1006) GO TO 23                          RET
    2 IF(JSTALL.EQ.1) GO TO 12                                          RET
      DO 11 I=1,IS                                                      RET
      IF(IBUF(2).EQ.LSTUDY(I)) GO TO 12                                 RET
   11 CONTINUE                                                          RET
      GO TO 10                                                          RET
   12 IF(JDTALL.EQ.1.AND.JOB(2).EQ.4HVOLU) GO TO 44                     RET
C                                                                       RET
C CHECK FOR SIMULTANEOUS @ALL@ AND DATE @SAVE@ OPTION                   RET
C                                                                       RET
      IF(JDTALL.EQ.1.AND.(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH))GO TO 41RET
      DO 13 I=1,ID                                                      RET
      IF(IBUF(3).EQ.LDATE(I)) GO TO 14                                  RET
   13 CONTINUE                                                          RET
      GO TO 10                                                          RET
   41 MDSAV = IBUF(3)                                                   RET
      GO TO 44                                                          RET
C                                                                       RET
C CHECK FOR DATE SAVE OPTION                                            RET
C                                                                       RET
   14 IF(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH) MDSAV=LDATE(I)           RET
   44  IF(MANALL.EQ.1) GO TO 16                                         RET
      DO 15 I=1,IM                                                      RET
      IF(IBUF(4).EQ.LMAN(I)) GO TO 16                                   RET
   15 CONTINUE                                                          RET
      GO TO 10                                                          RET
   16 IF(JTPALL.EQ.1) GO TO 18                                          RET
      DO 17 I=1,ITP                                                     RET
      IF(IBUF(5).EQ.LTYPE(I)) GO TO 18                                  RET
   17 CONTINUE                                                          RET
      GO TO 10                                                          RET
C                                                                       RET
C CHECK FOR MASTER SAMPLE NUMBER                                        RET
C                                                                       RET
   18 IF(IBUF(14).EQ.0) GO TO 21                                        RET
      MIH = IH-1                                                        RET
      DO 20 I=1,MIH                                                     RET
      IF(LHIT(I).EQ.IBUF(14)) GO TO 10                                  RET
   20 CONTINUE                                                          RET
      LHIT(IH)=IBUF(14)                                                 RET
C     PRINT 220,(LHIT(I),I=1,10)                                        RET
C 220 FORMAT(5X,5HLHIT2,5X,10I10)                                       RET
      GO TO 22                                                          RET
   21 LHIT(IH)=IBUF(1)                                                  RET
C     PRINT 221,(LHIT(I),I=1,10)                                        RET
C 221 FORMAT(5X,5HLHIT1,5X,10I10)                                       RET
```

```
C                                                                       RET
C  IF DATE SAVE OPTION WAS SPECIFIED FILL DATE ARRAY CORRESPONDING      RET
C     TO SAMPLE NUMBER HIT ARRAY                                        RE
C                                                                       RE
   22 IF(JOB(2).EQ.4HDATE.OR.JOB(2).EQ.4HBOTH) MDATE(IH)=MDSAV          RET
C                                                                       RET
C  IF TOTAL VOLUME SAVE OPTION WAS SPECIFIED FILL VOLUME ARRAY          RET
C         CORRESPONDING TO SAMPLE NUMBER HIT ARRAY                      RET
C                                                                       RET
      IF(JOB(2).EQ.4HVOLU.OR.JOB(2).EQ.4HBOTH)MTVOL(IH)=IBUF(10)        RET
       IH=IH+1                                                          RET
      IF(IH.LE.151)  GO TO 10                                           XREDUC
   23 NSMP = IH - 1                                                     RET
      IF(NSMP.LE.150)  GO TO 35                                         XREDUC
      PRINT 30                                                          RET
   30 FORMAT(1H1,17HHIT FILE OVERFLOW    )                              RET
      STOP 30                                                           RET
C                                                                       RET
   35 CALL RETRVE                                                       RET
      PRINT 300, LSTUDY,LMAN,JOB,LTYPE,NSMP,NDATA                       RET
  300 FORMAT(1H0,6HLSTUDY,10I10,//,2X,4HLMAN,9I10,//,                   DEC10
     .   1X, 4HJOB ,A4,5X,A4,5X,14I6,//,6H LTYPE,20I5,//,              RET
     .   1X, 7HNSMP = ,I5,10X,8HNDATA = ,I5)                           DEC10
      PRINT 301,(LDATE(I),I=1,ID)                                       RET
  301 FORMAT(6H LDATE /,(10I10))                                        RET
      PRINT 302,(LTEST(I),I=1,ITT)                                      RET
  302 FORMAT(6H LTEST  ,(10I10))                                        RET
      PRINT 306,(MTVOL(I),I=1,NSMP)                                     RET
  306 FORMAT(1H0,5HMTVOL,/,(10I10))                                     DE  )0
      PRINT 303,(MDATE(I),I=1,NSMP)                                     RE
  303 FORMAT(1H0,5HMDATE,/,(10I10))                                     DEC10
      PRINT 304,(LHIT(I),I=1,NSMP)                                      RET
  304 FORMAT(/ ,6H LHIT ,/,(10I10))                                     DEC10
      PRINT 305,(XDATA(I),I=1,NDATA)                                    RET
  305 FORMAT(1H0,5HXDATA,/,(8F15.6))                                    DEC10
C     PRINT 300, LSTUDY,LMAN,JOB,LTYPE,NSMP,NDATA                       RET
C 300 FORMAT(1H1,6HLSTUDY,10I10,//,2X,4HLMAN,9I10,//,                   RET
C    .       4HJOB ,A4,5X,A4,5X,14I6,//,5HLTYPE,20I5,//,                RET
C    .       7HNSMP = ,I5,10X,8HNDATA = ,I5,// )                        RET
C     PRINT 301,(LDATE(I),I=1,ID)                                       RET
C 301 FORMAT(5HLDATE /,(10I10))                                         RET
C     PRINT 302,(LTEST(I),I=1,ITT)                                      RET
C 302 FORMAT(5HLTEST  ,(10I10))                                         RET
C     PRINT 306,(MTVOL(I),I=1,NSMP)                                     RET
C 306 FORMAT(1H1,5HMTVOL,/,(10I10))                                     RET
C     PRINT 303,(MDATE(I),I=1,NSMP)                                     RET
C 303 FORMAT(1H1,5HMDATE,/,(10I10))                                     RET
C     PRINT 304,(LHIT(I),I=1,NSMP)                                      RET
C 304 FORMAT(//,5HLHIT ,/,(10I10))                                      RET
C     PRINT 305,(XDATA(I),I=1,NDATA)                                    RET
C 305 FORMAT(1H1,5HXDATA,/,(8F15.6))                                    RET
C                                                                       RET
C                                                                       RET
C     ------------------------------------------------------------------RET
      RETURN                                                            RE
C     CALL STAT                                                         RE
 4321 CONTINUE                                                          RET
      IFFF = 0                                                          RET
C                                                                       RET
C     ------------------------------------------------------------------RET
```

```
C                                                                       RET
C     REWIND 1                                                          RET
      CALL REWND(FITRA1)                                                RET
      GO TO 201                                                         RET
C                                                                       RET
  888 PRINT 8800                                                        RET
 8800 FORMAT(15H FILE 1 REWOUND)                                        RET
      RETURN                                                            RET
   99 CONTINUE                                                          RET
      CALL CLOSEM(FITRA1)                                               RET
      CALL CLOSEM(FITRA2)                                               RET
      CALL CLOSEM(FITRA3)                                               RET
      CALL CLOSEM(FITRA4)                                               RET
      IF(IOPLOT) 199,199,999                                            RET
  999 CALL STOPPLT                                                      RET
      PRINT 900                                                         RET
  900 FORMAT(19HONORMAL END OF JOB.)                                    RET
      STOP 2000                                                         RET
  199 STOP 1                                                            RET
      END                                                               RET
      SUBROUTINE SETCRI(LBUF,JNDX,LALL)                                 SETCRI
C  THIS SUBROUTINE STORES INFORMATION FROM ONE DATA CARD INTO THE        SETCRI
C                         CORRESPONDING CRITERIA ARRAY                  SETCRI
C                                                                       SETCRI
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)          SETCRI
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)           XREDUC
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),   SETCRI
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA           SETCRI
      COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEY,KYE,      SETCRI
     +                MFUNC,IFILE,IERR                                  SETCRI
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)           SETCRI
      DIMENSION LBUF(1)                                                 SETCRI
      IF(ISAV(2).EQ.4HALL .OR.ISAV(2).EQ.4H    ) GO TO 1                SETCRI
      JNDX = JNDX + 1                                                   SETCRI
      LBUF(JNDX)=ISAV(3)                                                SETCRI
      IF(ISAV(2).EQ.4HEACH) GO TO 3                                     SETCRI
      INXT=ISAV(3)+1                                                    SETCRI
    2 IF(INXT.GT.ISAV(4)) RETURN                                        SETCRI
      JNDX = JNDX + 1                                                   SETCRI
      LBUF(JNDX)=INXT                                                   SETCRI
      INXT=INXT+1                                                       SETCRI
      GO TO 2                                                           SETCRI
    3 DO 5 I=4,16                                                       SETCRI
      IF(ISAV(I).EQ.0) RETURN                                           SETCRI
      JNDX = JNDX + 1                                                   SETCRI
      LBUF(JNDX)=ISAV(I)                                                SETCRI
    5 CONTINUE                                                          SETCRI
      RETURN                                                            SETCRI
    1 LALL=1                                                            SETCRI
      RETURN                                                            SETCRI
      END                                                               SETCRI
      SUBROUTINE RETRVE                                                 RETRVE
C                                                                       RETRVE
C THIS SUBROUTINE RETREVES THOSE DATA VALUES FROM THE BLOOD OR          RETRVE
C      URINE FILES WHICH SATISFY THE INPUT CRITERIA                     RETRVE
C      THE DATA VALUES ARE STORED IN ARRAY XDATA                        RETRVE
C                                                                       RETRVE
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP        XREDUC
C     COMMON /INDEX/INDX(124),INDX1(124),INDX2(248),INDX3(124)          RETRVE
      COMMON/INDEX/INDX(136),INDX3(136)                                 DEC10
```

```
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)            XREDUC
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),    RETRVE
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA            RETRVE
      COMMON /SISBUF/ IBUF(35),IBL(125),IUR(35),IBUFF(30),KEE,KYE,       RETRVE
     +                MFUNC,IFILE,IERR                                   RETRVE
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)            RETRVE
      DIMENSION XTEMP(125)                                               RETRVE
      INTEGER KEY(2)                                                     RETRVE
      EQUIVALENCE(T,XTEMP)                                               RETRVE
      EQUIVALENCE (KEE,KEY(2))                                           RETRVE
      DATA(INDX(I),I=1,136)/400,401,402,403,404,405,503,406,407,408,409,DEC10
     .  410,                                                             DEC10
     1  411,412,413,414,415,416,417,418,419,420,421,422,423,424,425,426,DEC10
     2  427,428,429,430,431,432,433,434,435,436,437,438,439,440,441,442,DEC10
     3  443,444,445,446,447,448,449,450,451,452,453,454,455,456,457,458,DEC10
     4  459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,DEC10
     5  475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,490,DEC10
     6  491,492,493,494,495,496,497,498,499,500,501,502,300,301,302,303,DEC10
     7  304,305,306,307,321,308,309,310,311,312,313,314,315,316,317,318,DEC10
     .  319,                                                             DEC10
     8  320,514,515,516,517,518,519,520,521,522,523/                    DEC10
      DATA (INDX3(I),I=1,136)/2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,  DEC10
     1  18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38, DEC10
     2  39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59, DEC10
     3  60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80, DEC10
     4  81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,   DEC10
     5101,102,103,104,105,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,DEC10
     6  20,21,22,23,116,117,118,119,120,121,122,123,124,125/            DEC10
      DATA ICDT/445B/                                                    RET
      M=1                                                                RET
      KSAV=0                                                             RETRVE
      IERR=0                                                             RETRVE
      DO 27 K=1,NSMP                                                     RETRVE
      DO 27 I=1,ITT                                                      RETRVE
C                                                                        RETRVE
C  IDENTIFY TEST NUMBER IN INDEX                                         RETRVE
C                                                                        RETRVE
      DO 24 J=1,136                                                      DEC10
      IF(LTEST(I).EQ.INDX(J)) GO TO 25                                   DEC10
   24 CONTINUE                                                           DEC10
      STOP @INVALID TEST NUMBER@                                         DEC10
   25 IJ=J                                                               DEC10
      LREC=INDX3(IJ)                                                     DEC10
      IF(LTEST(I) -400) 41,31,31                                         RETRVE
   31 INDX1 = 1RU                                                        RETRVE
      GO TO 50                                                           RETRVE
   41 INDX1 = 1RB                                                        RETRVE
C                                                                        RETRVE
C  PICK UP KEY FROM HIT ARRAY OF SAMPLE NUMBERS                          RETRVE
C                                                                        RETRVE
   50 KEY(2)=LHIT(K)                                                     RETRVE
C                                                                        RETRVE
C DETERMINE IF DATA NEEDED IS FROM CURRENT RECORD                        RETRVE
C                                                                        RETRVE
      IF(KEY(2).EQ.KSAV) GO TO 26                                        RET E
C                                                                        RET E
C  CHECK FOR URINE OR BLOOD FILE                                         RETRVE
C                                                                        RETRVE
C  29 IF(INDX1(IJ).EQ.1RU) GO TO 28                                      RETRVE
   29 IF(INDX1.EQ.1RU) GO TO 28                                          RETRVE
```

```
C      CALL RANDOM(1,@RA3@,XTEMP,32,KEY,100 ,100 ,ICODE,LOC)                RETRVE
       CALL GET(FITRA3)                                                     RETRVE
       LOC = 3                                                              RETRVE
       IFILE = 3                                                            RETRVE
       IF(IERR.NE.0) GO TO 100                                              RETRVE
       GO TO 26                                                             RETRVE
C   28 CALL RANDOM(1,@RA2@,XTEMP,125,KEY,100 ,100 ,ICODE,LOC)               RETRVE
    28 CALL GET(FITRA2)                                                     RETRVE
       LOC = 2                                                              RETRVE
       IFILE = 2                                                            RETRVE
       IF(IERR.NE.0) GO TO 100                                              RETRVE
C                                                                           RETRVE
C                                                                           RETRVE
C  FETCH WORD NUMBER OF DATA RECORD FROM INDEX, STORE DATA                  RETRVE
C     VALUE IN CURRENT XDATA LOCATION AND INCREMENT                         RETRVE
C           XDATA LOCATION COUNTER M                                        RETRVE
C                                                                           RETRVE
C                                                                           RETRVE
C   26 LREC=INDX3(IJ)                                                       RETRVE
    26 XDATA(M)=XTEMP(LREC)                                                 RETRVE
    30 M=M+1                                                                RETRVE
       IF(M.GT.150)  GO TO 110                                              XREDUC
       KSAV=KEY(2)                                                          RETRVE
    27 CONTINUE                                                             RETRVE
C                                                                           RETRVE
C                                                                           RETRVE
C  NDATA = NUMBER OF DATA VALUES RETREIVED                                  RETRVE
C                                                                           RETRVE
C                                                                           RETRVE
       NDATA=M-1                                                            RETRVE
       RETURN                                                               RETRVE
   100 IF(IERR .EQ.ICDT) GO TO 105                                          RETRVE
       PRINT 101,IERR ,KEY(2)                                               RETRVE
   101 FORMAT(1H ,19HRANDOM ERROR-RETRVE,08,5X,I4)                          RETRVE
       STOP 100                                                             RETRVE
   105 PRINT 104,KEY(2)                                                     RETRVE
   104 FORMAT(1H ,27HSAMPLE NO. NOT FOUND-RETRVE,5X,I4)                     RETRVE
       XDATA(M)=-1.                                                         RETRVE
       IERR=0                                                               RETRVE
       GO TO 30                                                             RETRVE
   110 PRINT 111                                                            RETRVE
   111 FORMAT(1H1,43HDATA BUFFER FULL BEFORE ALL DATA RETREIVED    )        RETRVE
       STOP 110                                                             RETRVE
       END                                                                  RETRVE
       SUBROUTINE STAT                                                      STAT
       COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP           XREDUC
       COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)              XREDUC
       COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),      STAT
      .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA              STAT
       COMMON/RG/ DATESC(3,5),IVV(4), SSMEAN(3), SSIGMA(3), NNN(3),SSE(3)   STAT
       COMMON CHEM(200),TIME(200),NOB(3),TREAT(6),SL(6),SS(12),XMSC(12)     ANAC1
      2    ,F(12),PROBF(12),XM(6),YM(6),NDFC(12),SXX(6),SYY(6),SXY(6)      ANAC1
       COMMON ZCH(2,60),WORK(2,60)                                          ANAC1
       DIMENSION C(28)                                                      ANAC1
       REAL KSTUDY, KTYPE, KMAN, KTEST, NITS                                STAT
       DIMENSION ISIG(2)                                                    STAT
       DIMENSION X(1)                                                       STAT
       DIMENSION LTEXT(120), MTEXT(120), XOUT(120)                          STAT
       DIMENSION NDF(3), S(3), P(3), TM(3), WTV(3), XMS(3), IPP(3),IP2(3)   STAT
       DIMENSION ID1(120), ID2(120), LK(3)                                  STAT
```

```
      EQUIVALENCE(X,XDATA)                                                  STAT
      DATA ID1/120*1/, LK/3*0/                                              STAT
      DATA IPP/4HPRE  , 4HIN   , 4HPOST/                                    STAT
      DATA IP2/8HPRE-IN   , 8HIN-POST , 8HPRE-POST/                         STAT
C                                                                           STAT
C     FILL OUTPUT ARRAY                                                     STAT
C                                                                           STAT
      DATA ISTAR/1H*/, IBLANK/1H /, ISAVE/4HSAVE/                           STAT
      DATA IPLUS/1H+/, IMINUS/1H-/, IPPL/2H++/, IMM1/2H--/                  STAT
      DATA IDATE/4HDATE/, IVOLU/4HVOLU/, IBOTH/4HBOTH/                      STAT
      NOBS = NDATA                                                          STAT
    1 IF(NOBS - 1) 999,999,2                                                STAT
    2 LOOP = ITT                                                            STAT
      IDATEF = 0                                                            STAT
      IVOLUF = 0                                                            STAT
      IF(JOB(2).EQ.IDATE) IDATEF = 3                                        STAT
      IF(JOB(2).EQ.IVOLU) IVOLUF = 3                                        STAT
      IF(JOB(2).NE.IBOTH) GO TO 3                                           STAT
      IDATEF = 3                                                            STAT
      IVOLUF = 3                                                            STAT
    3 KKK = NOBS/LOOP                                                       STAT
      CALL TEXT(LSTUDY(1), LTYPE(1), LMAN(1), LTEST(1),                     STAT
     X KSTUDY, KTYPE, KMAN, KTEST, NITS)                                    STAT
      CALL DATE(IWORD)                                                      STAT
      PRINT 1000, JOB(3)                                                    STAT
 1000 FORMAT(1H1,20X,42HNASA MSC ENDOCRINE DATA RETREIVAL PROGRAM.,17/)     STAT
      PRINT 500, LSTUDY(1), KSTUDY,IWORD                                    STAT
  500 FORMAT(7HOSTUDY,,I2,2X, A8,20X,A10)                                   STAT
      PRINT 600, LTEST(1), KTEST,LMAN(1), KMAN                              STAT
  600 FORMAT(6HOTEST,,I3,2X A8 ,20X,4HMAN,,I4,2X, A8)                       STAT
      PRINT 700,(LTYPE(I),I=1,ITP)                                          STAT
  700 FORMAT(6HOTYPE,,20I3)                                                 DEC10
      PRINT 800, NITS                                                       DEC10
  800 FORMAT(15HOJULIAN DATE     , A8)                                      STAT
C     DO 150 L=1,LOOP                                                       STAT
      L = 1                                                                 STAT
      LOOP = 1                                                              STAT
      DO 210 II=LOOP,NOBS,LOOP                                              STAT
      I = II + L - LOOP                                                     STAT
      IPLACE = II/LOOP                                                      STAT
  210 XOUT(IPLACE) = XDATA(I)                                               STAT
C                                                                           STAT
C     SORT BY DATE                                                          STAT
C                                                                           STAT
      NOSLTS = NOBS/LOOP                                                    STAT
      NM1 = NOSLTS - 1                                                      STAT
      DO 230 I=1,NM1                                                        STAT
      IMAX = 9999                                                           STAT
      ISUB = I                                                              STAT
      DO 220 J=I,NOSLTS                                                     STAT
      IF(IMAX-MDATE(J)) 220,220,211                                         STAT
  211 IMAX = MDATE(J)                                                       STAT
      ISUB = J                                                              STAT
  220 CONTINUE                                                              STAT
      ITEMP = MDATE(ISUB)                                                   STAT
      TEMP = XOUT(ISUB)                                                     STAT
      MDATE(ISUB) = MDATE(I)                                                STAT
      XOUT(ISUB) = XOUT(I)                                                  STAT
      MDATE(I) = ITEMP                                                      STAT
      XOUT(I) = TEMP                                                        STAT
```

```
  230 CONTINUE                                                          STAT
      IG = LSTUDY(1)                                                    STAT
      IVV(1)=IVV(2)=IVV(3)=0                                            APR24
      IVV(4) = NOSLTS                                                   STAT
      DO 240 I=1,NOSLTS                                                 STAT
      IF(MDATE(I).LE.DATESC(2,IG)) IVV(2) = I                           STAT
      IF(MDATE(I).LE.DATESC(3,IG)) IVV(3) = I                           STAT
  240 XDATA(I) = XOUT(I)                                                STAT
      DO 133 I=1,3                                                      STAT
      NFIRST = IVV(I) + 1                                               STAT
      NLAST = IVV(I + 1)                                                STAT
      CALL MESIG(LTEXT,NFIRST,NLAST,SMEAN,SIGMA,NN)                     STAT
      SSMEAN(I) = SMEAN                                                 STAT
      SSIGMA(I) = SIGMA                                                 STAT
      NNN(I) = NN                                                       STAT
  133 CONTINUE                                                          STAT
      NFIRST = 1                                                        STAT
      NLAST = NOSLTS                                                    STAT
      CALL MESIG(LTEXT,NFIRST,NLAST,SMEAN,SIGMA,NN)                     STAT
C                                                                       STAT
C     IF THERE ARE 2 OR MORE PRE-FLIGHT OBSERVATIONS THE EXTREME VALUES STAT
C     ARE FLAGED ON THE BASIS OF PRE-FLIGHT MEAN AND STD. DEV.          STAT
C     OTHERWISE THE TOTAL MEAN AND STD. DEV. ARE USED.                  STAT
C                                                                       STAT
      IF(NNN(1) - 2) 134,136,136                                        STAT
  134 SSG = SIGMA                                                       STAT
      SSM = SMEAN                                                       STAT
      GO TO 137                                                         STAT
  136 SSG = SSIGMA(1)                                                   STAT
      SSM = SSMEAN(1)                                                   STAT
  137 T01 = 2.3263*SSG                                                  STAT
      T05 = 1.6449*SSG                                                  STAT
      DO 166 II=1,3                                                     STAT
      IFIRST = IVV(II)+ 1                                               STAT
      ILAST = IVV(II + 1)                                               STAT
      IF(IFIRST-ILAST)162,163,166                                       APR24
  163 PRINT 401                                                         APR24
  401 FORMAT(*  SINGLE VALUE NOT ANALYZED*)                             APR24
  162 DO 165 I=IFIRST,ILAST                                             STAT
      MTEXT(I) = IBLANK                                                 STAT
      IF(X(I)) 164,165,153                                              STAT
  153 DELTA = X(I) - SSM                                                STAT
      IF(DELTA) 155,154,157                                             STAT
  154 IF(X(I).NE.2.0) GO TO 164                                         STAT
      DELTA = SSM                                                       STAT
      GO TO 156                                                         STAT
  155 DELTA = -DELTA                                                    STAT
  156 IF(DELTA.GE.T05) MTEXT(I) = IMINUS                                STAT
      IF(DELTA.GE.T01) MTEXT(I) = IMM1                                  STAT
      GO TO 164                                                         STAT
  157 IF(DELTA.GE.T05) MTEXT(I) = IPLUS                                 STAT
      IF(DELTA.GE.T01) MTEXT(I) = IPPL                                  STAT
  164 PRINT 400,MDATE(I), X(I), LTEXT(I), MTEXT(I)                      STAT
  165 CONTINUE                                                          STAT
      PRINT 900,SSMEAN(II), SSIGMA(II), NNN(II)                         STAT
  166 CONTINUE                                                          STAT
      IF(JOB(3) - 1000) 150,145,145                                     STAT
  145 IF(NN) 150,150,146                                                STAT
  146 CALL STPLOT(LOOP,NOBS,NSLOT,IWORD,KSTUDY,KTYPE,KMAN,KTEST,SMEAN,  STAT
     X SIGMA)                                                           STAT
```

```
  150 CONTINUE                                                          STAT
C                                                                       STAT
C      ICASE = 0     NO TESTS                                           STAT
C      ICASE = 1     PREFLIGHT ONLY                                     STAT
C      ICASE = 2     PRE AND INFLIGHT ONLY                              STAT
C      ICASE = 3     ALL                                                STAT
C      ICASE = 4     PRE AND POSTFLIGHT ONLY                            STAT
C                                                                       STAT
       IF(NNN(1)-1) 170,170,171                                         DEC10
  170 NT = 0                                                            STAT
       ICASE = 0                                                        STAT
       GO TO 999                                                        DEC10
  171 SSE(1) = SSIGMA(1)/SQRT(FLOAT(NNN(1)))                            STAT
       PRINT 3000,NNN(1), SSMEAN(1), SSIGMA(1), SSE(1)                  STAT
 3000 FORMAT(20(4H----)/23H PARAMETRIC STATISTICS.//8HOSUMMARY/         STAT
     X39HOSAMPLE  N    MEAN         SD          SE/4HOPKT, I6, 3F11.3) STAT
       IF(NNN(2)-1) 172,172,176                                         DEC10
  172 IF(NNN(3)-1) 173,173,174                                          DEC10
  173 NT = 1                                                            STAT
       ICASE = 1                                                        STAT
       GO TO 999                                                        DEC10
  174 SSE(3) = SSIGMA(3)/SQRT(FLOAT(NNN(3)))                            STAT
       PRINT3010,NNN(3), SSMEAN(3), SSIGMA(3), SSE(3)                   STAT
       NT = 2                                                           STAT
       ICASE = 4                                                        STAT
       NN=NN-NNN(2)                                                     FIX2
C                                                                       STAT
       GO TO 180                                                        DEC10
C      CHANGE THIS LATER                                                STAT
C                                                                       STAT
  176 SSE(2) = SSIGMA(2)/SQRT(FLOAT(NNN(2)))                            STAT
       PRINT3020,NNN(2), SSMEAN(2), SSIGMA(2), SSE(2)                   STAT
 3020 FORMAT(4H0IN ,I6,3F11.3)                                          STAT
       IF(NNN(3)-1) 177,177,178                                         DEC10
  177 NT = 2                                                            STAT
       ICASE = 2                                                        STAT
       NN=NN-NNN(3)                                                     FIX2
       GO TO 180                                                        STAT
  178 SSE(3) = SSIGMA(3)/SQRT(FLOAT(NNN(3)))                            STAT
       PRINT3010,NNN(3), SSMEAN(3), SSIGMA(3), SSE(3)                   STAT
 3010 FORMAT(5HOPOST,I5,3F11.3)                                         STAT
       NT = 3                                                           STAT
       ICASE = 3                                                        STAT
  180 CONTINUE                                                          STAT
       SE = SIGMA/SQRT(FLOAT(NN))                                       STAT
       PRINT 3030, NN, SMEAN, SIGMA, SE                                 STAT
 3030 FORMAT(4H ---,10(4H----)/4H SUM,I6,3F11.3)                        STAT
C                                                                       STAT
C      GET RID OF NEGATIVE VALUES                                       STAT
C         I  RANGES OVER ALL POINTS.                                    STAT
C         J  RANGES OVER VALID POINTS.                                  STAT
C                                                                       STAT
       J = 1                                                            STAT
       DO 330 I=1,NOSLTS                                                STA
       IF(X(I)) 310,330,320                                             STA
  310 IF(X(I) + 2.0) 330,315,330                                        STA
  315 X(J) = 0.0                                                        STA
       GO TO 321                                                        ANAC1
  320 X(J) = X(I)                                                       STAT
  321 XOUT(J)=MDATE(I)                                                  ANAC1
```

```
      J = J + 1                                                          STAT
  330 CONTINUE                                                           STAT
C     SET FLAG FOR ANACOVA                                               ANAC1
      IF(LMAN(1).EQ.1) IANC=1                                            ANAC1
      IF(ICASE.EQ.2.OR.ICASE.EQ.3) GO TO 322                             ANAC1
      PRINT 605                                                          ANAC1
  605 FORMAT(*  MISSING INFLIGHT DATA - NO ANACOVA POSSIBLE*)            ANAC1
      IANC=0                                                             ANAC1
      GO TO 716                                                          ANAC1
C     PICK INFLIGHT DATA ONLY INTO CHEM AND TIME                         ANAC1
C     X IS CHANGED HERE AND LATER                                        ANAC1
  322 IF(LMAN(1).EQ.1) LOBS=0                                            ANAC1
      I=NNN(1)+1                                                         ANAC1
      J=NNN(1)+NNN(2)                                                    ANAC1
      DO 714 K=I,J                                                       ANAC1
      LOBS=LOBS+1                                                        ANAC1
      CHEM(LOBS)=X(K)                                                    ANAC1
      TIME(LOBS)=XOUT(K)                                                 ANAC1
  714 CONTINUE                                                           ANAC1
      NOB(LMAN(1))=NNN(2)                                                ANAC1
  716 CONTINUE                                                           FIX
      IF(ICASE.EQ.4) 710,711                                             FIX
  710 IF(NNN(2).EQ.1) GO TO 712                                          FIX
      GO TO 713                                                          FIX
  712 NNN1=NNN(1)+1                                                      FIX
      NSLTM1=J-1                                                         FIX
      DO 179 I=NNN1,NSLTM1                                               FIX
  179 X(I)=X(I+1)                                                        FIX
  713 NNN(2)=NNN(3)                                                      FIX
      NNN(3)=0                                                           FIX
  711 CONTINUE                                                           STAT
      CALL ACRDAN(X, NT, NNN, TM, WTV, S, GM, NDF, IER)                  STAT
      N1 = NDF(1)                                                        STAT
      N2 = NDF(2)                                                        STAT
      S1 = S(1)                                                          STAT
      S2 = S(2)                                                          STAT
      CALL FFOUT(S1,S2,N1,N2,FVAL,PRFVAL)                                STAT
      DO 193 IMS=1,3                                                     STAT
  193 XMS(IMS) = S(IMS)/NDF(IMS)                                         STAT
      PRINT 3090                                                         STAT
 3090 FORMAT(1X)                                                         STAT
      PRINT 3100                                                         STAT
 3100 FORMAT(28H0ANALYSIS OF VARIANCE TABLE,/35H0        DF       SS     STAT
     X  MS      F /35H        ----     ----       ----    ---)          STAT
      PRINT 3200, NDF(1), S(1), XMS(1), FVAL                             STAT
 3200 FORMAT(6H TREAT, I4, F10.1, F10.1, F8.3)                           STAT
      PRINT 3300, NDF(2), S(2), XMS(2)                                   STAT
 3300 FORMAT(6H0ERROR, I4, F10.1, F10.1, 6H  --- /4H ---, 9(4H----))     STAT
      PRINT 3400, NDF(3), S(3), XMS(3)                                   STAT
 3400 FORMAT(6H TOTAL, I4, F10.1, F10.1, 6H  --- )                       STAT
      PRINT 3800, FVAL, PRFVAL                                           STAT
 3800 FORMAT(1H0,F7.3, 23H IS SIGNIFICANT AT THE  ,F5.1, 14H PERCENT LEVSTAT
     1EL)                                                                STAT
      IF(NT-2) 999,850,824                                               DEC10
  824 IF(PRFVAL - 5.0) 825,825,850                                       STAT
  825 N1 = 1                                                             STAT
      PRINT 3500                                                         STAT
 3500 FORMAT(14H0CONTRAST   F )                                          STAT
      P(1) = 1                                                           STAT
      P(2) = -1                                                          STAT
```

```
      P(3)=0                                                                    FIX
      CALL ACTRST(TM,  NNN, 3, 1, P, 1, Q, SQ)                                  STAT
      CALL FFOUT(SQ,S2,N1,N2,FVAL,PRFVAL)                                       STAT
      PRINT 3600,IP2(1),FVAL,PRFVAL                                             FIX
 3600 FORMAT(1H0,A8,F8.3,23H IS SIGNIFICANT AT THE   ,F5.1,                     FIX
     X15H PERCENT LEVEL.)                                                       STAT
      P(1) = 0                                                                  STAT
      P(2) = 1                                                                  STAT
      P(3) = -1                                                                 STAT
      CALL ACTRST(TM,  NNN, 3, 1, P, 1, Q, SQ)                                  STAT
      CALL FFOUT(SQ,S2,N1,N2,FVAL,PRFVAL)                                       STAT
      PRINT 3600,IP2(2),FVAL,PRFVAL                                             FIX
      P(1) = 1                                                                  STAT
      P(2) = 0                                                                  STAT
      CALL ACTRST(TM,  NNN, 3, 1, P, 1, Q, SQ)                                  STAT
      CALL FFOUT(SQ,S2,N1,N2,FVAL,PRFVAL)                                       STAT
      PRINT 3600,IP2(3),FVAL,PRFVAL                                             FIX
  850 CONTINUE                                                                  STAT
  851 NP = 0                                                                    STAT
      DO 855 I=1,NT                                                             STAT
      NCDC = NNN(I)                                                             STAT
      DO 855 J=1,NCDC                                                           STAT
      NP = NP + 1                                                               STAT
  855 ID2(NP) = I                                                               STAT
      IF(ICASE.EQ.4) GO TO 860                                                  DEC1
      PRINT 4000                                                                STAT
 4000 FORMAT(1H0/20(4H----)/27H NON-PARAMETRIC STATISTICS.)                     STAT
C     CALL  KRUSWAL(X,ID1,ID2,NP,0,0,NT,NNN,LK)                                STAT
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,NT,NNN,LK,LK)                               FIX2
      PRINT 1100,IP2(1)                                                         ST
 1100 FORMAT(1H0/10H0CONTRAST , A8)                                             STA
      NP = NNN(1) + NNN(2)                                                      STAT
C     CALL  KRUSWAL(X,ID1,ID2,NP,0,0, 2,NNN,LK)                                STAT
      IPI=2                                                                     FIX2
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,2,NNN,LK,IPI)                               FIX2
      IF(ICASE.EQ.2) GO TO 998                                                  FIX
      NP = NNN(2) + NNN(3)                                                      STAT
      IST = NNN(1) + 1                                                          STAT
      PRINT 1100,IP2(2)                                                         STAT
C     CALL  KRUSWAL(X(IST),ID1(IST),ID2(IST),NP,0,0,2,NNN(2),LK)                STAT
      IPI=3                                                                     FIX2
      CALL KRUSWAL(X(IST),ID1(IST),ID2(IST),NP,0,1,2,NNN(2),LK,IPI)             FIX2
      NCDC = NNN(3)                                                             STAT
      DO 880 J=1,NCDC                                                           STAT
      K = NNN(1) + J                                                            STAT
      L=NNN(1)+NNN(2)+J                                                         FIX
      X(K) = X(L)                                                               STAT
      ID1(K) = ID1(L)                                                           STAT
  880 ID2(K) = ID2(L)                                                           STAT
      NP=NNN(1)+NNN(3)                                                          FIX
      NNN(2)=NNN(3)                                                             FIX
C     CALL  KRUSWAL(X,ID1,ID2,NP,0,0, 2,NNN,LK)                                STAT
  860 PRINT 1100,IP2(3)                                                         DEC10
      IPI=3                                                                     FIX2
      IF(ICASE.EQ.4) IPI=2                                                      FIX
      CALL KRUSWAL(X,ID1,ID2,NP,0,1,2,NNN,LK,IPI)                               FI
  998 CONTINUE                                                                  FIX
      IF(LMAN(1).LT.3.OR.IANC.NE.1) GO TO 999                                   ANAC1
      PRINT 1000,JOB(3)                                                         ANAC1
      PRINT 604,LSTUDY(1),KSTUDY,IWORD,LTEST(1),KTEST,LTYPE(1)                  ANAC1
```

```
604 FORMAT(7HOSTUDY ,I2,2X,A8,2X,A10,2X,5HTEST ,I3,2X,A8,2X,5HTYPE ,   ANAC1
   2I3,*  INFLIGHT DATA FOR ANACOVA*)                                  ANAC1
    PRINT 602                                                          ANAC1
    K=0                                                                ANAC1
    DO 715 L=1,3                                                       ANAC1
    J=K+1                                                              ANAC1
    K=K+NOB(L)                                                         ANAC1
    PRINT 602                                                          ANAC1
602 FORMAT(1H )                                                        ANAC1
    M=0                                                                ANAC1
    DO 717 I=J,K                                                       ANAC1
    M=M+1                                                              ANAC1
717 PRINT 601,M,CHEM(I),TIME(I)                                        ANAC1
715 CONTINUE                                                           ANAC1
601 FORMAT(I4,F12.4,F5.0)                                              ANAC1
    NT=3                                                               ANAC1
    IOPT=2                                                             ANAC1
    CALL ANACOVA(CHEM,TIME,LOBS,NT,NOB,TREAT,SL,SS,XMSC,F,PROBF,       ANAC1
   2     IOPT,XM,YM,SXX,SYY,SXY,NDFC)                                  ANAC1
    IF(IOPT.LT.0) PRINT 603                                            ANAC1
603 FORMAT(@  ANACOVA  ERROR**********@)                               ANAC1
    J=1                                                                ANAC1
    K=0                                                                ANAC1
    DO 719 LAS=1,3                                                     ANAC1
    K=K+NOB(LAS)                                                       ANAC1
    L=0                                                                ANAC1
    DO 718 I=J,K                                                       ANAC1
    L=L+1                                                              ANAC1
    ZCH(2,L) =0.                                                       ANAC1
    ZCH(1,L)=CHEM(I)-(YM(LAS)+SL(LAS)*(TIME(I)-XM(LAS)))               ANAC1
    CHEM(I)=ZCH(1,L)                                                   ANAC1
    WORK(1,L)=0.                                                       ANAC1
    WORK(2,L)=0.                                                       ANAC1
718 CONTINUE                                                           ANAC1
    CALL LAGCOR(NOB(LAS),CHEM(J),NOB(LAS),CHEM(J),C,1,28)              ANAC1
    PRINT 610,LAS                                                      ANAC1
610 FORMAT(*0 MAN*,I3,*  LAG CORRELATIONS*/*0SEQ    C-VALUE*)          ANAC1
    PRINT 611,(I,C(I),I=1,28)                                          ANAC1
611 FORMAT(I6,G15.6)                                                   ANAC1
    N=NOB(LAS)                                                         ANAC1
    CALL FOURG(ZCH,N,-1,WORK)                                          ANAC1
    DO 723 I=1,N                                                       ANAC1
    WORK(1,I)=ZCH(1,I)**2+ZCH(2,I)**2                                  ANAC1
723 WORK(2,I)=ATAN2(ZCH(2,I),ZCH(1,I))                                 ANAC1
    PRINT 602                                                          ANAC1
    PRINT 607                                                          ANAC1
607 FORMAT(1H0,T18,*FAST FOURIER TRANSFORMS*/* SEQ OMEGA*,             ANAC1
   1       6X,@REAL          IMAGINARY       R**2+I**2      ARCTAN(I/R)@) ANAC1
    DO 720 M=1,N                                                       ANAC1
    QNM=FLOAT(N)/FLOAT(M-1)                                            ANAC1
    IF(M.EQ.1) QNM=0.                                                  ANAC1
    PRINT 606,M,QNM,ZCH(1,M),ZCH(2,M),WORK(1,M),WORK(2,M)              ANAC1
720 CONTINUE                                                           ANAC1
606 FORMAT(1X,I3,F6.2,4G15.6)                                          ANAC1
    J=K+1                                                              ANAC1
719 CONTINUE                                                           ANAC1
999 PRINT 1000,JOB(3)                                                  DEC10
    RETURN                                                             DEC10
200 FORMAT(10X,F12.4,2X,2A1)                                           STAT
300 FORMAT(13HOGRAND MEAN =,F10.3,  8H     SD =,F10.3,7H     N =,I3)   STAT
```

```
  400 FORMAT(I10,F12.4,2X,A4,2X,A2)                                      STAT
  900 FORMAT( 7HOMEAN =,F10.3,  8H     SD =,F10.3,7H     N =,I3//)       STAT
      END                                                                ST
      SUBROUTINE MESIG(LTEXT,NFIRST,NLAST,SMEAN,SIGMA,NN)                ME
      DIMENSION LTEXT(1)                                                 MESIG
      NN = 0                                                             MESIG
      SUM = 0.0                                                          MESIG
      SSQ = 0.0                                                          MESIG
      IF(NFIRST-NLAST) 10,10,90                                          APR24
   10 DO 140 I=NFIRST,NLAST                                              MESIG
      NSLOT = 1                                                          MESIG
      CALL OTEST(1,SUM,SSQ,NN,LTEXT)                                     MESIG
  140 CONTINUE                                                           MESIG
      IF(NN.EQ.0) GO TO 90                                               DEC10
      SMEAN = SUM/NN                                                     MESIG
      SIGMA=0.                                                           DEC10
      IF(NN.EQ.1) RETURN                                                 DEC10
      IF((NN*SSQ-SUM*SUM).LE.0.) RETURN                                  DEC10
      SIGMA = ((NN*SSQ - SUM*SUM)/(NN*(NN - 1)))**0.5                    MESIG
      RETURN                                                             MESIG
   90 SMEAN = 0.0                                                        MESIG
      SIGMA = 0.0                                                        MESIG
      RETURN                                                             MESIG
      END                                                                MESIG
      SUBROUTINE OTEST(I,SUM,SSQ,NN,LTEXT)                              OTEST
      DIMENSION LTEXT(1)                                                 OTEST
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP         XREDUC
      DATA IZ/4H0.00/, ISS/4HSKIP/, INV/4HINVL/                          OTEST
      DATA IBLANK/4H    /                                                JAN
      IF(XDATA(I)) 310,410,510                                           OTEST
  310 IXT = -XDATA(I) + 0.5                                              OTEST
      IF(IXT - 2) 315,312,315                                            OTEST
  312 LTEXT(I) = IZ                                                      OTEST
      NN = NN + 1                                                        OTEST
      RETURN                                                             OTEST
  315 LTEXT(I) = ISS                                                     OTEST
      RETURN                                                             OTEST
  410 LTEXT(I) = INV                                                     OTEST
      RETURN                                                             OTEST
  510 NN = NN + 1                                                        OTEST
      LTEXT(I) = IBLANK                                                  OTEST
      SUM = SUM + XDATA(I)                                               OTEST
      SSQ = SSQ + XDATA(I)*XDATA(I)                                      OTEST
      RETURN                                                             OTEST
      END                                                                OTEST
      SUBROUTINE UERTST                                                  UERTST
      RETURN                                                             UERTST
      END                                                                UERTST
      SUBROUTINE  STPLOT(LOOP,NOBST,NSLOT,IWORD,KSTUDY,KTYPE,KMAN,KTEST,STPLOT
     X SMEAN,SIGMA)                                                      STPLOT
      COMMON XDATA(150),T(200),IOPLOT,IFFF,KOT,KSKIP,LT,LM,LSTOP         XREDUC
      COMMON/HITBLK/LHIT(150),MDATE(150),ISAV (16),MTVOL(150)            XREDUC
      COMMON /HEADER/ LSTUDY(10),IS,LDATE(150),ID,LMAN(9),IM,JOB(16),    STPLOT
     .                LTYPE(20),ITP,LTEST(125),ITT,NSMP,NDATA            STPLOT
      DIMENSION YDATE(200),XP(3)                                         REDUCE
      COMMON/RG/ DATESC(3,5),IVV(4), SSMEAN(3), SSIGMA(3), NNN(3)        STPLOT
C     REAL*8 MSTUDY,MTYPE,MMAN,MTEST                                     STPLOT
      REAL   MSTUDY,MTYPE,MMAN,MTEST                                     STPLOT
      REAL KSTUDY, KTYPE, KMAN, KTEST, NITS                              STPLOT
      DIMENSION NSMAN(7)                                                 STPLOT
```

```
      EQUIVALENCE(YDATE,T)                                                  STPLOT
      DATA NSMAN/4,9,8,0,1,2,3/                                             STPLOT
      IF(NOBST - 1) 911,911,1                                               STPLOT
    1 MFIRST = MDATE(1)                                                     STPLOT
      DO 10 II=1,NOBST                                                      STPLOT
      IF(MDATE(II) - MFIRST) 9,10,10                                        STPLOT
    9 MFIRST = MDATE(II)                                                    STPLOT
   10 YDATE(II) = MDATE(II)                                                 STPLOT
      J1 = NOBST + 1                                                        STPLOT
      J2 = NOBST + 2                                                        STPLOT
      CALL SCALE(XDATA, 3.0,NOBST,1)                                        STPLOT
C     XDATA(NOBST + 1) = 0.0                                                STPLOT
      IF(XDATA(J2)) 911,911,11                                              STPLOT
   11 IG = LSTUDY(1)                                                        STPLOT
      YDATE(NOBST + 1) = DATESC(1,IG)                                       STPLOT
      YDATE(NOBST + 2) = 24.0                                               STPLOT
      UP = (DATESC(2,IG) - DATESC(1,IG))/YDATE(NOBST + 2)                   STPLOT
      DOWN=(DATESC(3,IG) - DATESC(1,IG))/YDATE(NOBST + 2)                   STPLOT
      XP(1) = 0.1                                                           STPLOT
      XP(2) = 0.1+ UP                                                       STPLOT
      XP(3) = 0.1+ DOWN                                                     STPLOT
      NTIC = 3                                                              STPLOT
      CALL TEXT(LSTUDY(1), LTYPE(1), LMAN(1), LTEST(1),                     STPLOT
     X KSTUDY, KTYPE, KMAN, KTEST, NITS)                                    STPLOT
      KS = KSKIP - KSKIP/3*3                                                STPLOT
      IF(KS) 208,108,208                                                    STPLOT
C 108 CALL PLOT(4.25,10.5,-3)                                               STPLOT
  108 CALL PLOT(8.50,10.5,-3)                                               STPLOT
      FPN = LTYPE(1)                                                        STPLOT
      CALL SYMBOL(0.30,-.4,0.10,KSTUDY    , 0.0, 8)                         STPLOT
      CALL SYMBOL(1.20,-.4,0.10, KTEST    , 0.0, 8)                         STPLOT
      CALL SYMBOL(2.00,-.4,0.10,4HTYPE, 0.0, 4)                             STPLOT
      CALL NUMBER(2.40,-.4,0.10,FPN,0.0,-1)                                 STPLOT
      CALL SYMBOL(2.60,-.4,0.10,IWORD,0.0,10)                               STPLOT
  208 CALL PLOT(0.0,-3.5,-3)                                                STPLOT
      CALL AXIS(0.0,0.0,12H JULIAN DATE,                                    STPLOT
     1   -12, 4.0,00.0,YDATE(J1),YDATE(J2),-1,4,3)                          STPLOT
C     CALL AXIS (X,Y,BCD,NC,SIZE,THETA,YMIN,DY,NDEC,NLAB,NTIC)              STPLOT
C     CALL AXIS(0.0,0.0,16H      DATA VALUE,16,3.0,  90.0, Y(J1),Y(J2))     STPLOT
      CALL AXIS(0.0,0.0,NITS    , 8,3.0,  90.0, XDATA(J1),                  STPLOT
     1 XDATA(J2), -1,  4, 1)                                                STPLOT
      FPN = LMAN(1)                                                         STPLOT
      CALL SYMBOL(0.10,3.01,0.07,KMAN, 0.0, 3)                              STPLOT
      DO 308 JJ=1,3                                                         STPLOT
      KG = 4 - JJ                                                           STPLOT
      CALL NUMBER(XP(KG), 2.89, 0.07, SSMEAN(KG), 0.0, 2)                   STPLOT
  308 CALL NUMBER(XP(KG), 2.77, 0.07, SSIGMA(KG), 0.0, 2)                   STPLOT
      CALL PLOT(UP,0.0,3)                                                   STPLOT
      CALL PLOT(UP,3.0,2)                                                   STPLOT
      CALL PLOT(DOWN,3.0,3)                                                 STPLOT
      CALL PLOT(DOWN,0.0,2)                                                 STPLOT
  708 LL = NSMAN(LMAN(1))                                                   STPLOT
      CALL LINE(YDATE,XDATA,NOBST,1, 0,LL)                                  STPLOT
  811 CONTINUE                                                              STPLOT
      KSKIP = KSKIP + 1                                                     STPLOT
  911 RETURN                                                                STPLOT
      END                                                                   STPLOT
      SUBROUTINE TEXT(LSTUDY,LTYPE,LMAN,LTEST,NSTUDY,NTYPE,NMAN,NTEST,      TEXT
     1 NITS)                                                                TEXT
C     REAL*8 KSTUDY,MSTUDY(6),KTEST,MURN(124),MBLD(24),MITS(124),           TEXT
```

```
 REAL   KSTUDY,MSTUDY(5),KTEST,MURN(124),MBLU(24),MITS(124),          TEXT
1MITSP(24),KTYPE,MTYPE(15),MM,KITS                                    TEXT
 REAL NSTUDY, NTYPE, NMAN, NTEST, KMAN, MMAN(9), MITS                 TEXT
 DATA MSTUDY/8HSMEAT    ,8HSKYLAB 2,8HSKYLAB 3,8HSKYLAB 4,8HAPOLLO17TEXT
                                                                      TEXT
1/                                                                    TEXT
 DATA MMAN/4HCDR ,4HPLT ,4HSPT ,4HC1  ,4HC2  ,4HC3  ,4HC4  ,          TEXT
X4HC5  ,4HC6  /                                                       TEXT
 DATA (MURN(I),I=1,107)                                               TEXT
X           /8HEPI      ,8HNOREPI   ,8HADH      ,8HHYDRO    ,8HALDO     ,TEXT
X8H17OH     ,8HOSMO     ,8HNA       ,8HK        ,8HMG       ,8HPO4      ,TEXT
X8HCA       ,8HCL       ,8HFE       ,8HSP.GR.   ,8HCREAT    ,8HURICACID,TEXT
X8HB        ,8HSI       ,8HFF       ,8HAL       ,8HMO       ,8HCU       ,TEXT
X8HZN       ,8HTI       ,8HNI       ,8HSR       ,8HCR       ,8HBL       ,TEXT
X8HMN       ,8HLI       ,8HRB       ,8HPD       ,8HAND      ,8HETIO     ,TEXT
X8HUREA     ,8H11=0 AND,8H11=0ETIO,8H11OH AND,8H11OHETIO,8HTOTAL    ,TEXT
X8HLYS      ,8HHIS      ,8HCH3      ,8HARG      ,8HHYP      ,8HASP      ,TEXT
X8HTHR      ,8HSER      ,8HGLU      ,8HPRO      ,8HGLY      ,8HALA      ,TEXT
X8HCYS/2    ,8HVAL      ,8HMET      ,8HILE      ,8HLEU      ,8HTYR      ,TEXT
X8HPHE      ,8HHLYS     ,8HGAMMA-AB,8HORN       ,8HETH      ,8HNH3      ,TEXT
X8HLYS      ,8H1-CH3HIS,8HHIS       ,8H3-CH3HIS,8HANS       ,8HTRY      ,TEXT
X8HCRE      ,8HCAR      ,8HARG      ,8HPSER     ,8HPETN     ,8HTYR      ,TEXT
X8HUREA     ,8HHYP      ,8HASP      ,8HTHR      ,8HSER      ,8HASP NH2  ,TEXT
X8HGLU NN2  ,8HSAR      ,8HPRO      ,8HGLU      ,8HCIT      ,8HGLC NH2  ,TEXT
X8HGLY      ,8HALA      ,8HALPHA-AA,8HALPHA-AB,8HVAL      ,8HCYS/2    ,TEXT
X8HCYT      ,8HMET      ,8HILE      ,8HLEU      ,8HTYR      ,8HPHE      ,TEXT
X8HB-ALA    ,8HB-AIB    ,8H5HIAA    ,8H         ,8H         ,8H         /JAN25
 DATA (MURN(I),I=108,124)/                                            TEXT
X8H         ,8H         ,8H         ,8H         ,8H         ,8H         ,TEXT
X8H         ,8H         ,8HCALORIES,8HPROTEIN ,8HDIET CA ,8HDIET P  ,TEXT
X8HDIET NA  ,8HDIET MG  ,8HDIET K   ,8HDIET H2O,8HWEIGHT  /           TEXT
 DATA MBLD /8HHYDRO    ,8HALDO     ,8HHGH      ,8HANGIO    ,8HINSULIN  ,TEXT
X8HT4       ,8HACTH     ,8HADH      ,8HPTH      ,8HCAL      ,8HVIT.D.   ,TEXT
X8HTSH      ,8HOSMO     ,8HNA       ,8HK        ,8HMG       ,8HPO4      ,TEXT
X8HCA       ,8HCL       ,8HGLU      ,8HTP       ,8HTESTOS   ,8H         ,JAN25
X8H         /                                                         TEXT
 DATA (MITS(I),I=1,107)                                               TEXT
X           /8HUG/TV    ,8HUG/TV    ,8HMU/TV    ,8HUG/TV    ,8HUG/TV    ,TEXT
X8HMG/TV    ,8HMOSM/TV  ,8HMEQ/TV   ,8HMEQ/TV   ,8HMEQ/TV   ,8HMG/TV    ,TEXT
X8HMEQ/TV   ,8HMEQ/TV   ,8HML0      ,8H         ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HUG/ML    ,8HUG/ML    ,8HUG/ML    ,8HUG/ML    ,8HUG/ML    ,8HUG/ML    ,TEXT
X8HUG/ML    ,8HUG/ML    ,8HUG/ML    ,8HUG/ML    ,8HUG/ML    ,8HUG/ML    ,TEXT
X8HUG/ML    ,8HUG/ML    ,8HUG/ML    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HTMG/TV   ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMMG/TV   ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMMG/TV   ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HMMG/TV   ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,TEXT
X8HMG/TV    ,8HMG/TV    ,8HMG/TV    ,8H         ,8H         ,8H         /JAN25
 DATA (MITS(I),I=108,124)/                                            TEXT
X8H         ,8H         ,8H         ,8H         ,8H         ,8H         ,TEXT
X8H         ,8H         ,8HKCAL     ,8HGRAMS    ,8HMG       ,8HMG       ,TEXT
X8HMG       ,8HMG       ,8HMG       ,8HML       ,8HLBS.     /           TEXT
 DATA MITSB/8HUG/100ML,8HPG/ML     ,8HNG/ML    ,8HMUG/ML/H,8HUU/ML     ,TEXT
X8HUG/100ML,8H          ,8H         ,8HPG/ML    ,8HPG/ML    ,8HNG/ML    ,TEXT
```

```
     X8H         ,8HMOSM/L   ,8HNEQ/L    ,8HNEQ/L    ,8HMG PC    ,8HMG PC    ,JAN25
     X8HMG PC    ,8HMG PC    ,8HMG PC    ,8HMG PC    ,8HUG/TV    ,8H         ,JAN25
     X8H         /                                                           TEXT
      NSTUDY = MSTUDY(LSTUDY)                                                TEXT
      NMAN = MMAN(LMAN)                                                      TEXT
      IF(LTEST - 400) 10,20,20                                               TEXT
   10 KTEST = MBLD(LTEST - 299)                                              TEXT
      KITS = MITSB(LTEST-299)                                                TEXT
      GO TO 30                                                               TEXT
   20 KTEST = MURN(LTEST - 399)                                              TEXT
      KITS = MITS(LTEST-399)                                                 TEXT
   30 NTEST = KTEST                                                          TEXT
      NITS = KITS                                                            TEXT
      RETURN                                                                 TEXT
      END                                                                    TEXT
      SUBROUTINE KRUSWAL(XIN,ID1,ID2IN,N,NCEN,JOP,K,NU,NC,IP2)               SETWRK
C     THIS PROGRAM CALCULATES  A  GENERALIZED KRUSKAL-WALLIS K-SAMPLE        KRUSWAL
C     TEST. WHEN K=2, THE TEST IS EQUAL TO GEHAN'S GENERALIZED WILCOXON      KRUSWAL
C     TEST. CALCULATIONS FOLLOW THE PROCEDURES GIVEN BY MANTEL (BIOMETRIC    KRUSWAL
C     MARCH 1967)                                                            KRUSWAL
C                                                                            KRUSWAL
C     IN CASE OF TWO-SAMPLE, IF THE SAMPLE SIZES ARE DIFFERENT, IT IS        KRUSWAL
C     SUGGESTED TO NAME THE ONE WITH LESS OBSERVATIONS SAMPLE 1.              KRUSWAL
C      CALL NKRUSWAL(X,ID1,ID2,NALL,NC,JOP,NCOND,NUNCEN,NCEN)                KRUSWAL
C                                                                            KRUSWAL
C      X(I),I=1,NALL     ARRAY CONTAINING ALL OBS IN THE SAMPLES             KRUSWAL
C       ID1(I)=1         ITH OBS IS UNCENSORED                               KRUSWAL
C             =0         ITH OBS IS CENSORED                                 KRUSWAL
C       ID2(I)=J         ITH OBS IS FROM JTH SAMPLE                          KRUSWAL
C       NALL             TOTAL NO OF OBS                                     KRUSWAL
C       NC               TOTAL NO OF CENSORED OBS                            KRUSWAL
C       JOP=1            PRINT INFO                                          KRUSWAL
C          =0            DO NOT PRINT                                        KRUSWAL
C       NCOND            NO OF SAMPLES                                       KRUSWAL
C       NUNCEN(I)        NO OF UNCENSORED OBS IN SAMPLE I                    KRUSWAL
C       NCEN(I)          NO OF CENSORED OBS IN SAMPLE I                      KRUSWAL
      DIMENSION XIN(1),ID1(1),ID2IN(1),NU(6),NC(6),XY(120),ID2(120)          SETWRK
      DIMENSION R1( 600),R2( 600)                                            KRUSWAL
C                                                                            KRUSWAL
C     MAXIMUM NO. OF SAMPLES = 6.                                            KRUSWAL
C     TOTAL NUMBER OF OBSERVATIONS ALLOWED = 600.                            KRUSWAL
C     K = NO. OF SAMPLES                                                     KRUSWAL
C                                                                            KRUSWAL
C      MOVE INCOMMING ARRAYS TO WORK ARRAYS                                  SETWRK
      DO 10 I=1,N                                                            SETWRK
      XY(I)=XIN(I)                                                           SETWRK
      ID2(I)=ID2IN(I)                                                        SETWRK
   10 CONTINUE                                                               SETWRK
C                                                                            KRUSWAL
C     ORDER OBS. IN ASCENDING ORDER                                          KRUSWAL
C                                                                            KRUSWAL
      CALL SORT2(XY,N,ID1,ID2)                                               KRUSWAL
C     COMPUTATION OF R1                                                      KRUSWAL
C       STEPS 1 AND 2 : RANK FROM LEFT TO RIGHT, OMITTING RIGHT CENSORED     KRUSWAL
C                       VALUES. ASSIGN NEXT HIGHER RANK TO RIGHT             KRUSWAL
C                       CENSORED VALUES                                      KRUSWAL
C                                                                            KRUSWAL
      IRANK=0                                                                KRUSWAL
      DO 90I=1,N                                                             KRUSWAL
      IF (ID1(I).EQ.0) GO TO 101                                             KRUSWAL
```

```
      IRANK=IRANK+1                                                     KRUSWAL
      R1(I)=IRANK                                                       KRUSWAL
      GO TO 90                                                          KRUSWAL
  101 R1(I)=IRANK+1                                                     KRUSWAL
   90 CONTINUE                                                          KRUSWAL
C                                                                       KRUSWAL
C       STEP 3 : REDUCE  THE RANK OF TIED OBSERVATIONS TO THE LOWEST    KRUSWAL
C                RANK FOR THE VALUE                                     KRUSWAL
C                                                                       KRUSWAL
      K1=N-1                                                            KRUSWAL
      L1=1                                                              KRUSWAL
   12 IF (XY(L1).NE.XY(L1+1)) GO TO 11                                  KRUSWAL
      JEMP=ID1(L1)*ID1(L1+1)                                            KRUSWAL
      IF(JEMP .EQ. 0) GO TO 11                                          KRUSWAL
      R1(L1+1)=R1(L1)                                                   KRUSWAL
      IF (L1.EQ.K1) GO TO 13                                            KRUSWAL
      L1=L1+1                                                           KRUSWAL
      GO TO 12                                                          KRUSWAL
   11 IF (L1.EQ.K1) GO TO 13                                            KRUSWAL
      L1=L1+1                                                           KRUSWAL
      GO TO 12                                                          KRUSWAL
   13 CONTINUE                                                          KRUSWAL
C                                                                       KRUSWAL
C     COMPUTATION OF R2                                                 KRUSWAL
C       STEP 1 : RANK FROM RIGHT TO LEFT                                KRUSWAL
C                                                                       KRUSWAL
      DO 14 I=1,N                                                       KRUSWAL
   14 R2(I)=N-I+1                                                       KRUSWAL
C                                                                       KRUSWAL
C       STEP 2 : REDUCE THE RANK OF TIED OBSERVATIONS TO THE LOWEST RANKKRUSWAL
C                FOR THE VALUE                                          KRUSWAL
C                                                                       KRUSWAL
      L1=N                                                              KRUSWAL
   22 IF (XY(L1).NE.XY(L1-1)) GO TO 21                                  KRUSWAL
      JEMP=ID1(L1)*ID1(L1-1)                                            KRUSWAL
      IF (JEMP .EQ. 0) GO TO 21                                         KRUSWAL
      R2(L1-1)=R2(L1)                                                   KRUSWAL
      IF (L1.EQ. 2) GO TO 23                                            KRUSWAL
      L1=L1-1                                                           KRUSWAL
      GO TO 22                                                          KRUSWAL
   21 IF (L1.EQ. 2) GO TO 23                                            KRUSWAL
      L1=L1-1                                                           KRUSWAL
      GO TO 22                                                          KRUSWAL
   23 CONTINUE                                                          KRUSWAL
C                                                                       KRUSWAL
C       STEP 3 : REDUCE THE RANK OF RIGHT CENSORED OBSERVATIONS TO UNITYKRUSWAL
C                                                                       KRUSWAL
      IF (NCEN .EQ. 0) GO TO 501                                        KRUSWAL
      DO 24 I=1,N                                                       KRUSWAL
      IF (ID1(I) .EQ. 1) GO TO 24                                       KRUSWAL
      R2(I)=1.                                                          KRUSWAL
   24 CONTINUE                                                          KRUSWAL
C                                                                       KRUSWAL
C     COMPUTE FINAL SCORES -R1(I)                                       KRUSWAL
C                                                                       KRUSWAL
  501 CONTINUE                                                          KRUSWAL
      DO 25 I=1,N                                                       KRUSWAL
   25 R1(I)=R1(I)-R2(I)                                                 KRUSWAL
      IF(JOP.NE.1) GO TO 37                                             KRUSWAL
      PRINT 30                                                          KRUSWAL
```

```
 30   FORMAT(1H0,8X,@I@,8X,@OBSERVATIONS@,8X,@SAMPLE@,8X,@SCORES@)      KRUSWAL
      DO 31 I=1,N                                                      KRUSWAL
      IF (ID1(I).EQ.0) GO TO 34                                        KRUSWAL
      PRINT 33, I,XY(I),ID2(I),R1(I)                                   KRUSWAL
      GO TO 31                                                         KRUSWAL
   34 PRINT 35, I,XY(I),ID2(I),R1(I)                                   KRUSWAL
 33   FORMAT(1H ,6X,I3,7X,F8.1,15X,I2,12X,F6.0)                        KRUSWAL
 35   FORMAT(1H ,6X,I3,7X,F8.1,1H+,14X,I2,12X,F6.0)                    KRUSWAL
 31   CONTINUE                                                         KRUSWAL
   37 PRINT 36                                                         KRUSWAL
 36   FORMAT(1H0)                                                      KRUSWAL
      IF (K.GT.2) GO TO 200                                            KRUSWAL
      CALL TWOSPL(R1,ID2,N,NU,NC,IP2)                                  FIX2
      GO TO 1000                                                       KRUSWAL
200   CALL AKSPL(K,N,R1,NU,NC,ID2)                                     KRUSWAL
1000  CONTINUE                                                         KRUSWAL
      RETURN                                                           KRUSWAL
      END                                                              KRUSWAL
      SUBROUTINE SORT2(X,N,ID,IC)                                      SORT2
      DIMENSION X(1),ID(1),IC(1)                                       SORT2
      DO 1 I=1,N                                                       SORT2
      J=N-I+1                                                          SORT2
      JJ=J-1                                                           SORT2
      IF (JJ .LT. 1)GO TO 1                                            SORT2
      DO 2 K=1,JJ                                                      SORT2
      IF (X(K).NE.X(J)) GO TO 3                                        SORT2
      IF (ID(K)-ID(J)) 4,2,2                                           SORT2
 3    IF (X(K) .LT. X(J)) GO TO 2                                      SORT2
 4    X1=X(J)                                                          SORT2
      ITEM=ID(J)                                                       SORT2
      ICTE=IC(J)                                                       SORT2
      X(J)=X(K)                                                        SORT2
      ID(J)=ID(K)                                                      SORT2
      IC(J)=IC(K)                                                      SORT2
      X(K)=X1                                                          SORT2
      ID(K)=ITEM                                                       SORT2
      IC(K)=ICTE                                                       SORT2
 2    CONTINUE                                                         SORT2
 1    CONTINUE                                                         SORT2
      RETURN                                                           SORT2
      END                                                              SORT2
      SUBROUTINE TWOSPL(R1,ID2,N,NU,NC,IP2)                            FIX2
      DIMENSION ID2(1),R1(1),NU(1),NC(1)                               TWOSPL
C     DATA ONE,TWO/@ONE@,@TWO@/                                        TWOSPL
     .DATA ONE/10HONE        /                                         TWOSPL
      DATA TWO/10HTWO        /                                         TWOSPL
      WW=0.                                                            TWOSPL
      DO 26 I=1,N                                                      TWOSPL
      IF(ID2(I).EQ.IP2) GO TO 26                                       FIX2
      WW=WW+R1(I)                                                      TWOSPL
 26   CONTINUE                                                         TWOSPL
      SUM=0.                                                           TWOSPL
      DO 27 I=1,N                                                      TWOSPL
 27   SUM=SUM+R1(I)**2                                                 TWOSPL
      XN1=NU(1)+NC(1)                                                  TWOSPL
      XN2=NU(2)+NC(2)                                                  TWOSPL
      XN=N                                                             TWOSPL
      VAR=XN1*XN2*SUM/(XN*(XN-1.))                                     TWOSPL
      VAR=SQRT(VAR)                                                    TWOSPL
C                                                                      TWOSPL
```

```
C     CALCULATE CONTINUITY CORRECTION                                    TWOSPL
C                                                                        TWOSPL
      SIGN=-1.                                                           TWOSPL
      IF (WW.LT.0.) SIGN=1.                                              TWOSPL
      COR=.5                                                             TWOSPL
      IF (FLOAT((NC(1)+NC(2))/N) .LT. 0.2) COR=1.0                       TWOSPL
      COR=SIGN*COR                                                       TWOSPL
      IF (WW.EQ.0.) COR=0.                                               TWOSPL
C                                                                        TWOSPL
C     CALCULATE FINAL W SCORE AND PROBABILITY.                           TWOSPL
C                                                                        TWOSPL
      WSCORE=(WW+COR)/VAR                                                TWOSPL
      WSC=WSCORE                                                         TWOSPL
      IF (ABS(WSCORE)-3.1) 305,305,302                                   TWOSPL
  305 P1=100.*(1.-PROB(WSC))                                             TWOSPL
      GO TO 301                                                          TWOSPL
  302 P1=0.1                                                             TWOSPL
  301 CONTINUE                                                           TWOSPL
      P2=P1*2.                                                           TWOSPL
      P2=AMIN1(P2,100.)                                                  TWOSPL
      PRINT 600, WW,VAR,WSCORE                                           TWOSPL
  600 FORMAT(1H0,3X,1HW,7X,8HST. DEV.,3X,10HASYMPTOTIC/25X,6HWSCORE/1X,FTWOSPL
     19.0,F8.2,7X,F5.2)                                                  DEC10
      PRINT 601, WSCORE,P2,TWO                                           TWOSPL
      PRINT 601, WSCORE,P1,ONE                                           TWOSPL
  601 FORMAT(1X,F5.2,1X,@IS SIGNIFICANT AT THE @, F7.1,@ PERCENT LEVEL -TWOSPL
     1@,A3,@ TAILED TEST@)                                               TWOSPL
      PRINT 602                                                          TWOSPL
  602 FORMAT(1H0)                                                        TWOSPL
      RETURN                                                             TWOSPL
      END                                                                TWOSPL
      SUBROUTINE AKSPL(K,N,P1,NU,NC,ID2)                                 AKSPL
      DIMENSION R1(1),NU(1),NC(1),W(6),ID2(1)                            AKSPL
C                                                                        AKSPL
C     TEST STATISTIC FOR K(GREATER THAN 2)-SAMPLE CASE                   AKSPL
C                                                                        AKSPL
      DO 201 I=1,K                                                       AKSPL
  201 W(I)=0.                                                            AKSPL
      DO 202 IJ=1,N                                                      AKSPL
      I=ID2(IJ)                                                          AKSPL
  202 W(I)=W(I)+R1(IJ)                                                   AKSPL
      T=0.                                                               AKSPL
      DO 203 IJ=1,N                                                      AKSPL
  203 T=T+R1(IJ)**2                                                      AKSPL
      B=0.                                                               AKSPL
      PRINT 300                                                          AKSPL
  300 FORMAT(1H0,@SAMPLE@,8X,@W(I)@,10X,@N(I)@)                          AKSPL
      DO 204 I=1,K                                                       AKSPL
      UC=NU(I)+NC(I)                                                     AKSPL
      PRINT 301, I,W(I),UC                                               AKSPL
  301 FORMAT(1H ,3X,I1,7X,F8.0,7X,F5.0)                                  AKSPL
  204 B=B+(W(I)**2/UC)                                                   AKSPL
      PRINT 302, B,T                                                     AKSPL
  302 FORMAT(1H0,@B =@,F12.2,5X,@T =@,F10.0)                             AKSPL
      WSCORE=(B/T)*FLOAT(N-1)                                            AKSPL
C                                                                        AKSPL
C     WSCORE HAS CHI-SQUARE DISTRIBUTION WITH (K-1) D. F.                AKSPL
C                                                                        AKSPL
      WS=WSCORE                                                          AKSPL
      XK=K-1                                                             AKSPL
```

```
      XM=WS/2.                                                          AKSPL
      IC=XK/2.                                                          AKSPL
      SWS=SQRT(WS)                                                      AKSPL
      EM=1./EXP(XM)                                                     AKSPL
      PRINT 205, WSCORE                                                 AKSPL
  205 FORMAT(1H0,@WSCORE = @,F7.3)                                      AKSPL
      XK2=XK/2.                                                         AKSPL
      IF ((XK2-FLOAT(IC)).NE.0) GO TO 500                               AKSPL
      SUM=0.                                                            AKSPL
      PROD=1.                                                           AKSPL
      DO 1 I=1,IC                                                       AKSPL
      IF (I.GT.1) GO TO 2                                               AKSPL
      XI=1                                                              AKSPL
      GO TO 3                                                           AKSPL
    2 XI=I-1                                                            AKSPL
    3 PROD=XI*PROD                                                      AKSPL
      TERM=XM**(I-1)/PROD                                               AKSPL
    1 SUM=SUM+TERM                                                      AKSPL
      CPROB=SUM*EM                                                      AKSPL
      P=100.*CPROB                                                      AKSPL
      IF (P.LT.0.1000) GO TO 10                                         AKSPL
      PRINT 206, WSCORE,P                                               AKSPL
  206 FORMAT(1H0,F7.3,@ IS SIGNIFICANT AT THE @,F5.1,@ PERCENT LEVEL@)  AKSPL
      GO TO 1000                                                        AKSPL
   10 P=0.10                                                            AKSPL
      PRINT 12, WSCORE,P                                                AKSPL
   12 FORMAT(1H0,F7.3,@ IS SIGNIFICANT WITH PROBABILITY LESS THAN @,F7.3AKSPL
     1,@ PERCENT LEVEL@)                                                AKSPL
      GO TO 1000                                                        AKSPL
  500 GAMH=1.7724538509                                                 AKSPL
      TERM=SQRT(XM)/(0.5*GAMH)                                          AKSPL
      SUM=TERM                                                          AKSPL
      NR=(XK-3.)/2.                                                     AKSPL
      IF(NR.EQ.0) GO TO 502                                             AKSPL
      DO 501 I=1,NR                                                     AKSPL
      XI=I                                                              AKSPL
      TERM=TERM*2.*XM/(2.*XI+1.)                                        AKSPL
  501 SUM=SUM+TERM                                                      AKSPL
  502 CHISQ=(SUM*EM)*100.                                               AKSPL
      P2=100.*2.*(1.-PROB(SWS))                                         AKSPL
      P=P2+CHISQ                                                        AKSPL
      IF (P.LT.0.0005) GO TO 14                                         AKSPL
      PRINT 206,  WSCORE,P                                              AKSPL
      GO TO 1000                                                        AKSPL
   14 P=0.001                                                           AKSPL
      PRINT 12,  WSCORE,P                                               AKSPL
 1000 CONTINUE                                                          AKSPL
      RETURN                                                            AKSPL
      END                                                               AKSPL
      FUNCTION PROB(X)                                                  PROB
C                                                                       PROB
C       THIS FUNCTION ROUTINE COMPUTES                                  PROB
C                    DISTRIBUTION FUNCTION(X)        IF X GE 0          PROB
C                1 - DISTRIBUTION FUNCTION(X)        IF X LT 0          PROB
C       OF A ST. NORMAL VARIABLE USING APPROXIMATION 26.2.19  P.932     PROB
C       HANDBOOK OF MATH. FUNCTIONS                                     PROB
      DATA D1,D2,D3,D4,D5,D6/.0498673470,.0211410061,.0032776263,.000038PROB
     10036,.0000488906,.0000053830/                                     PROB
      IF (X) 20,30,40                                                   PROB
   30 PROB=0.5                                                          PROB
```

```
      RETURN                                                            PROB
 20   X1=-X                                                             PROB
      GO TO 50                                                          PROB
 40   X1=X                                                              PROB
 50   A=1.+X1*(D1+X1*(D2+X1*(D3+X1*(D4+X1*(D5+X1*D6)))))                PROB
      PROB=1.-0.5*A**(-16)                                              PROB
 60   CONTINUE                                                          PROB
      RETURN                                                            PROB
      END                                                               PROB
      SUBROUTINE FFOUT(S1, S2, N1, N2,FDAT, PRB)                        FFOUT
      FDAT = (S1/N1)/(S2/N2)                                            FFOUT
      PRB= FISH(FDAT,N1,N2)                                             FFOUT
      PRB = (1.0 - PRB)*100.                                            FFOUT
      IF(PRB - 0.1) 1,2,2                                               FFOUT
    1 PRB = 0.1                                                         FFOUT
    2 RETURN                                                            FFOUT
      END                                                               FFOUT
      FUNCTION FISH(F,N1,N2)                                            FISH
      LOGICAL E1,E2,E3                                                  FISH
       IF(N1.GE.100.AND.N2.GE.100) GOTO 9                               FISH
C-------------------------------------------------------------------------FISH
C     INITIALIZATION AND SETTING OF LOGICAL SWITCHES  TO .TRUE. IF      FISH
C     THE DEGREES OF FREEDOM ARE EVEN                                   FISH
C-------------------------------------------------------------------------FISH
      E1=.FALSE.                                                        FISH
      E2=.FALSE.                                                        FISH
      E3=.FALSE.                                                        FISH
      IF(MOD(N1,2).EQ.0) E1=.TRUE.                                      FISH
      IF(MOD(N2,2).EQ.0) E2=.TRUE.                                      FISH
      X=N2/(N2+N1*F)                                                    FISH
      IF(.NOT.(E1.OR.E2)) GO TO 5                                       FISH
      IF(E1.AND..NOT.E2) GO TO 1                                        FISH
      IF(.NOT.E1.AND.E2) GO TO 2                                        FISH
      IF(N1.LE.N2) GO TO 1                                              FISH
C-------------------------------------------------------------------------FISH
C   INITIALIZATION FOR SECOND DEGREE OF FREEDOM EVEN  AND LESS THAN     FISH
C      FIRST DEGREE OF FREEDOM IF IT TOO IS EVEN                        FISH
C-------------------------------------------------------------------------FISH
    2 I=N1                                                              FISH
      N1=N2                                                             FISH
      N2=I                                                              FISH
      X=1.0-X                                                           FISH
      E3=.TRUE.                                                         FISH
C-------------------------------------------------------------------------FISH
C     INITIALIZATION FOR FIRST DEGREE OF FREEDOM EVEN AND LESS THAN     FISH
C   SECOND DEGREE OF FREEDOM IF IT IS EVEN                              FISH
C-------------------------------------------------------------------------FISH
    1 Y=1.0-X                                                           FISH
C-------------------------------------------------------------------------FISH
C     CALCULATION OF PROBABILITY FOR AT LEAST ONE DEGREE OF FREEDOM EVENFISH
C-------------------------------------------------------------------------FISH
      FISH=0.0                                                          FISH
      H=SQRT(X**N2)                                                     FISH
      M=N1/2-1                                                          FISH
      MCDC = M + 1                                                      FISH
      DO 3 ICDC=1,MCDC                                                  FISH
      I = ICDC - 1                                                      FISH
      FISH=FISH+H                                                       FISH
    3 H=(H*Y*(N2+2.*I))/(2.*(I+1.))                                     FISH
      IF(E3) GO TO 4                                                    FISH
```

```
C-------------------------------------------------------------------FISH
C       ADJUST CALCULATED PROBABILITY IF ITS ONES COMPLEMENT WAS      FISH
C    CALCULATED ORIGINALLY                                           FISH
C-------------------------------------------------------------------FISH
      FISH=1.0-FISH                                                  FISH
      RETURN                                                         FISH
    4 I=N1                                                           FISH
      N1=N2                                                          FISH
      N2=I                                                           FISH
      RETURN                                                         FISH
C-------------------------------------------------------------------FISH
C     CALCULATION OF THE PROBABILITY FOR BOTH DEGREES OF FREEDOM ODD  FISH
C-------------------------------------------------------------------FISH
    5 Y=1.0-X                                                        FISH
      H=.63661977*SQRT(X*Y)                                          FISH
      FISH=.63661977*ACOS(SQRT(X))                                   FISH
      IF(N2.EQ.1) GO TO 8                                            FISH
      M=N2-2                                                         FISH
      DO  6  I=1,M,2                                                 FISH
      FISH=FISH+H                                                    FISH
    6 H=H*X*(I+1)/(I+2)                                              FISH
    8 IF(N1.EQ.1) RETURN                                             FISH
      H=H*N2                                                         FISH
      M=N1-2                                                         FISH
      DO  7  I=1,M,2                                                 FISH
      FISH=FISH-H                                                    FISH
    7 H=H*Y*(N2+I)/(I+2)                                             FISH
      RETURN                                                         FISH
    9  D1=N1                                                         FISH
       D2=N2                                                         FISH
       DT=(D1/D2)*F                                                  FISH
       DN=SQRT((2.*D2-1.)*DT)-SQRT(2.*D1-1.)                         FISH
       X=DN/SQRT(1.+DT)                                              FISH
       FISH=PHI(X)                                                   FISH
       RETURN                                                        FISH
      END                                                            FISH
      REAL FUNCTION PHI(X)                                           PHI
C-------------------------------------------------------------------PHI
C                                                                    PHI
C  PHI CALCULATES THE AREA UNDER THE NORMAL CURVE                    PHI
C  A TRANSFORMATION AND J-FRACTION ARE USED ( SEE METHOD )           PHI
C                                                                    PHI
C-------------------------------------------------------------------PHI
      LOGICAL UPPER                                                  PHI
      IF(X.LT.-13.27) GO TO 6                                        PHI
      IF(X.GT.8.5) GO TO 8                                           PHI
      IF(X.NE.0.0) GO TO 2                                           PHI
      PHI=0.50                                                       PHI
      RETURN                                                         PHI
    2 UPPER=X.GT.0.0                                                 PHI
      Z=    (ABS(X))                                                 PHI
      Y= 5.6418953027302E-1* EXP(- Z*Z /2.E0)                        PHI
      Z = Z /1.4142135623731E0                                       PHI
      T=0.0E0                                                        PHI
      IF( ABS(Y/Z).GT.0.0) T=Y/(Z+6.9183675618730E-6                 PHI
     1+5.0025350900390E-1/(Z+1.2386797611409E-2+7.7267300865878E-1   PHI
     2/(Z-4.3263982143053E0 +7.3456287718055E1/(Z+1.5040871364290E1  PHI
     3+6.2086245657235 6E0 /(Z+8.8971612130791E0 +4.9182171845874E1  PHI
     4/(Z-2.5108230069509E0 -2.8225972942737E0/(Z-9.7597917308472E-1 PHI
     5+2.4244213526837E1 /(Z+4.8008570125081E0 +4.9227853919002E-1   PHI
```

```
     6/(Z+7.662117092766 1E0 +5.028561912578 8E1/(Z-4.652928498465 5E0      PHI
     7  ))))))))))                                                         PHI
      T=T/2.E0                                                             PHI
      IF(UPPER) GO TO 4                                                    PHI
      PHI=T                                                                PHI
      RETURN                                                               PHI
    4 PHI=1.0E0-T                                                          PHI
      RETURN                                                               PHI
    6 PHI = 0.0                                                            PHI
      RETURN                                                               PHI
    8 PHI = 1.0                                                            PHI
      RETURN                                                               PHI
      END                                                                  PHI
      SUBROUTINE ERROR                                                     ERROR
      COMMON /SISBUF/ IBUF(35),IUR(125),IBL(35),IBUFF(30),KEE,KEF,         ERROR
     +                MFUNC,IFILE,IERR                                     ERROR
      COMMON /FF/ FITRA1(35),FITRA2(35),FITRA3(35),FITRA4(35)              ERROR
      GO TO(100,200,300,400) IFILE                                         ERROR
  100 IERR = IFETCH(FITRA1,3LIRS)                                          ERROR
      CALL STOREF(FITRA1,3LIRS,0)                                          ERROR
      GO TO 500                                                            ERROR
  200 IERR = IFETCH(FITRA2,3LIRS)                                          ERROR
      CALL STOREF(FITRA2,3LIRS,0)                                          ERROR
      GO TO 500                                                            ERROR
  300 IERR = IFETCH(FITRA3,3LIRS)                                          ERROR
      CALL STOREF(FITRA3,3LIRS,0)                                          ERROR
      GO TO 500                                                            ERROR
  400 IERR = IFETCH(FITRA4,3LIRS)                                          ERROR
      CALL STOREF(FITRA4,3LIRS,0)                                          ERROR
  500 IF(IERR.EQ.0) RETURN                                                 ERROR
      PRINT 601,IFILE,IERR,KEE,KEF                                         ERROR
      RETURN                                                               ERROR
  601 FORMAT(* RA*,I1,* ERROR *,O10,2I10)                                  ERROR
      END                                                                  ERROR
      SUBROUTINE ANACOVA(Y,X,L ,NT,NOB,TREAT,SL,SS,XMS,F,PROBF,IOPT,       ANACOVA
     Z                   XM,YM,SXX,SYY,SXY,NDF)                            ANACOVA
      DIMENSION Y(2),X(2),NOB(2),YM(2),XM(2),SL(2),SS(2),XMS(2),F(2),      ANACOVA
     2          PROBF(2),SXX(2),SYY(2),SXY(2),TREAT(2),NDF(2)              ANACOVA
      SS(1)=0                                                              ANACOVA
      SS(9)=0                                                              ANACOVA
      NTT=NT+1                                                             ANACOVA
      NT2=NT+2                                                             ANACOVA
      NT3=NT+3                                                             ANACOVA
      SS(12)=0                                                             ANACOVA
      L=0                                                                  ANACOVA
      DO 9 I=1,NT3                                                         ANACOVA
      SXX(I)=0                                                             ANACOVA
      SYY(I)=0                                                             ANACOVA
      SXY(I)=0                                                             ANACOVA
      XM(I)=0                                                              ANACOVA
      YM(I)=0                                                              ANACOVA
    9 CONTINUE                                                             ANACOVA
      DO 20 I=1,NT                                                         ANACOVA
      IF(NOB(I).EQ.0) GO TO 99                                             ANACOVA
      NO=NOB(I)                                                            ANACOVA
      DO 10 J=1,NO                                                         ANACOVA
      L=L+1                                                                ANACOVA
      XM(I)=XM(I)+X(L)                                                     ANACOVA
      YM(I)=YM(I)+Y(L)                                                     ANACOVA
      SXX(I)=SXX(I)+X(L)*X(L)                                              ANACOVA
```

```
      SYY(I)=SYY(I)+Y(L)*Y(L)                                          ANACOVA
   10 SXY(I)=SXY(I)+X(L)*Y(L)                                          ANACOVA
      YM(NTT)=YM(NTT)+YM(I)                                            ANACOVA
      XM(NTT)=XM(NTT)+XM(I)                                            ANACOVA
      SXX(NTT)=SXX(NTT)+SXX(I)                                         ANACOVA
      SXY(NTT)=SXY(NTT)+SXY(I)                                         ANACOVA
      SYY(NTT)=SYY(NTT)+SYY(I)                                         ANACOVA
      SS(12)=SS(12)+SYY(I)                                             ANACOVA
      XM(I)=XM(I)/NO                                                   ANACOVA
      YM(I)=YM(I)/NO                                                   ANACOVA
      SXX(I)= SXX(I)-NO*XM(I)**2                                       ANACOVA
      SYY(I)=(SYY(I)-NO*YM(I)**2)/(NO-1.)                              ANACOVA
      SXY(I)= SXY(I)-NO*XM(I)*YM(I)                                    ANACOVA
      SL(I)=SXY(I)/SXX(I)                                              ANACOVA
      TREAT(I)= YM(I)-SL(I)*XM(I)                                      ANACOVA
      SS(1)=SS(1)+NO*YM(I)*YM(I)                                       ANACOVA
      SS(9)=SS(9)+SXX(I)*SL(I)**2                                      ANACOVA
      SXY(NT3)=SXY(NT3)+NO*XM(I)*YM(I)                                 ANACOVA
      SXX(NT3)=SXX(NT3)+NO*XM(I)**2                                    ANACOVA
      SXX(NT2)=SXX(NT2)+SXX(I)                                         ANACOVA
      SXY(NT2)=SXY(NT2)+SXY(I)                                         ANACOVA
      SXX(I)=SXX(I)/(NO-1.)                                            ANACOVA
      SXY(I)=SXY(I)/(NO-1.)                                            ANACOVA
   20 CONTINUE                                                         ANACOVA
      YM(NTT)=YM(NTT)/L                                                ANACOVA
      XM(NTT)=XM(NTT)/L                                                ANACOVA
      SXX(NTT)=(SXX(NTT)-L*XM(NTT)*XM(NTT))/(L-1.)                     ANACOVA
      SXY(NTT)=(SXY(NTT)-L*XM(NTT)*YM(NTT))/(L-1.)                     ANACOVA
      SSU=L*YM(NTT)*YM(NTT)                                            ANACOVA
      SYY(NTT)=(SYY(NTT)-SSU)/(L-1.)                                   ANACOVA
      SL(NTT)=SXY(NTT)/SXX(NTT)                                        ANACOVA
      SS(NTT)=SS(12)                                                   ANACOVA
      SL(NT2)=SXY(NT2)/SXX(NT2)                                        ANACOVA
      SS(6)=SXX(NT2)*SL(NT2)**2                                        ANACOVA
      SXX(NT2)=SXX(NT2)/(L-NT)                                         ANACOVA
      SXY(NT2)=SXY(NT2)/(L-NT)                                         ANACOVA
      SS(8)=SS(9)-SS(6)                                                ANACOVA
      SXX(NT3)= (SXX(NT3) - L*XM(NTT)**2) /(NT-1.)                     ANACOVA
      SXY(NT3)= (SXY(NT3) - L*XM(NTT)*YM(NTT))/(NT-1.)                 ANACOVA
      IF(SXX(NT3).GT.0) GO TO 50                                       ANACOVA
      SL(NT3)=1.0E28                                                   ANACOVA
      TREAT(NT3)=YM(NTT)                                               ANACOVA
      SS(4)=0.                                                         ANACOVA
      SS(3)=0.                                                         ANACOVA
      GO TO 51                                                         ANACOVA
   50 CONTINUE                                                         ANACOVA
      SL(NT3)= SXY(NT3)/SXX(NT3)                                       ANACOVA
      SS(4)= (NT-1.)*SXX(NT3)*SL(NT3)**2                               ANACOVA
      SS(3)= (SL(NTT)-SL(NT3))**2/(1./(SXX(NT3)*(NT-1.))               ANACOVA
     2                            + 1./(SXX(NT2)*(L-NT)))               ANACOVA
      TREAT(NT3)=YM(NTT)-SL(NT3)*XM(NTT)                               ANACOVA
   51 CONTINUE                                                         ANACOVA
      SS(1)=SS(1)-SSU                                                  ANACOVA
      SS(5)= SS(1)-SS(4)                                               ANACOVA
      SS(2)= (L-1.)*SXX(NTT)*SL(NTT)**2                                ANACOVA
      SS(10)=SSU+SS(1)+SS(9)                                           ANACOVA
      SS(7) =SSU+SS(1)+SS(6)                                           ANACOVA
      TREAT(NTT)=YM(NTT)-SL(NTT)*XM(NTT)                               ANACOVA
      SS(11)=SS(12)-SS(10)                                             ANACOVA
      TREAT(NT2)=0                                                     ANACOVA
```

```
      NDF(1)=NT-1                                                       ANACOVA
      NDF(2)= 1                                                         ANACOVA
      NDF(3)= 1                                                         ANACOVA
      NDF(4)= 1                                                         ANACOVA
      NDF(5)= NT-2                                                      ANACOVA
      NDF(6)= 1                                                         ANACOVA
      NDF(7)= NT+1                                                      ANACOVA
      NDF(8)= NT-1                                                      ANACOVA
      NDF(9)= NT                                                        ANACOVA
      NDF(10)=2*NT                                                      ANACOVA
      NDF(12)=L                                                         ANACOVA
      NDF(11)=L-NDF(10)                                                 ANACOVA
      DO 55 K=1,11                                                      ANACOVA
      IF(NDF(K).GT.0) GO TO 54                                          ANACOVA
      XMS(K)=0                                                          ANACOVA
      GO TO 55                                                          ANACOVA
   54 XMS(K)=SS(K)/NDF(K)                                               ANACOVA
   55 CONTINUE                                                          ANACOVA
      DO 60 K=1,10                                                      ANACOVA
      F(K)=XMS(K)/XMS(11)                                               ANACOVA
      IF(F(K).GT.0.) GO TO 59                                           ANACOVA
      PROBF(K)=0.                                                       ANACOVA
      GO TO 60                                                          ANACOVA
   59 PROBF(K)=FISH(F(K),NDF(K),NDF(7))                                 ANACOVA
   60 CONTINUE                                                          ANACOVA
      IF(IOPT.LT.1) GO TO 999                                           ANACOVA
      WRITE(6,150)                                                      ANACOVA
      WRITE(6,1)                                                        ANACOVA
    1 FORMAT(@0 TREAT@,T10,@NUM@,T18,@INDEPENDENT@,T34,@INDEPENDENT@,   ANACOVA
     2        T50,@DEPENDENT@,T67,@DEPENDENT@,T83,@COVARIANCE@,         ANACOVA
     3        T99,@INTERCEPT@,T118,@SLOPE@/2X,@MENT@,T10,@BER@,         ANACOVA
     4        T20,@MEAN@,T35,@VARIANCE@,T51,@MEAN@,T67,@VARIANCE@//)    ANACOVA
      DO 70 I=1,NT                                                      ANACOVA
      WRITE(6,2) I,NOB(I),XM(I),SXX(I),YM(I),SYY(I),SXY(I),TREAT(I),    ANACOVA
     2           SL(I)                                                  ANACOVA
   70 CONTINUE                                                          ANACOVA
    2 FORMAT(2X,I5,1X,I5,7(4X,G13.6))                                   ANACOVA
      WRITE(6,4) SXX(NT2),SXY(NT2),SL(NT2)                              ANACOVA
    4 FORMAT(1X,124(1H-)/2X,@1SLOPE@,26X,G13.6,38X,G13.6,21X,G13.6)     ANACOVA
      WRITE(6,5) SXX(NT3),SXY(NT3),SL(NT3)                              ANACOVA
    5 FORMAT(/2X,@SLOPE @,26X,G13.6,38X,G13.6,21X,G13.6/2X,@OF MEANS@)  ANACOVA
      WRITE(6,3)    L    ,XM(NTT),SXX(NTT),YM(NTT),SYY(NTT),SXY(NTT),  ANACOVA
     Z              TREAT(NTT),SL(NTT)                                 ANACOVA
    3 FORMAT(/2X,@TOTAL@,1X,I5,7(4X,G13.6))                             ANACOVA
      WRITE(6,150)                                                      ANACOVA
      IF(IOPT.EQ.3) GO TO 80                                            ANACOVA
      WRITE(6,100) SSU,(NDF(I),SS(I),XMS(I),F(I),PROBF(I),I=1, 5)       ANACOVA
  100 FORMAT(//,15X,93(1H-),/,47X,@ANALYSIS OF COVARIANCE TABLE@,/,15X,  ANACOVA
     193(1H-),/,15X,@SOURCE OF@,9X,@DEGREES OF@,5X,@SUM OF@,10X,@MEAN@,  ANACOVA
     212X,@F-@,14X,@PROB@,/,15X,@VARIATION@,9X,@FREEDOM@,8X,@SQUARES@,   ANACOVA
     39X,@SQUARE@,10X,@RATIO@,11X,@F@,/,15X,93(1H-),/,15X,@MEAN@,19X,    ANACOVA
     *@1@,8X,G13.6,/,15X,@TREATMENTS@,4X,I10,8X,4(G13.6,3X),/,15X,       ANACOVA
     493(1H-),/,                                                        ANACOVA
     A            15X,@1 POPULATION @,1X,I10,8X,4(G13.6,3X)             ANACOVA
     B          ,/,15X,@1 POP.- SLOPE@,1X,I10,8X,4(G13.6,3X)            ANACOVA
     C          ,/,15X,@    OF MEANS @,/,15X ,                        ANACOVA
     * 93(1H-),/,15X,@SLOPE OF MEANS@,I10,8X,4(G13.6,3X)               ANACOVA
     D          ,/,15X,@DIFF MEAN SL.@,1X,I10,8X,4(G13.6,3X)            ANACOVA
     E          ,/,15X,@AND  PAR. SL.@                                  ANACOVA
     F          )                                                       ANACOVA
```

```
      WRITE(6,102)      (NDF(I),SS(I),XMS(I),F(I),PROBF(I),I=6,10),      ANACOVA
     2              NDF(11),SS(11),XMS(11),NDF(12),SS(12)                ANACOVA
  102 FORMAT(                                                            ANACOVA
     F          ,/,15X,@PARALLEL LINE@,1X,I10,8X,4(G13.6,3X)             ANACOVA
     G          ,/,15X,@   SLOPE     @,/, 15X,                          ANACOVA
     H 93(1H-),/,15X,@PARALLEL LINE@,1X,I10,8X,4(G13.6,3X)               ANACOVA
     I          ,/,15X,@   MODEL     @,/,  15X,                         ANACOVA
     J 93(1H-),/,15X,@NON-           @,1X,I10,8X,4(G13.6,3X)             ANACOVA
     K          ,/,15X,@  PARALLELISM@,                                  ANACOVA
     L          ,/,15X,@INDIVIDUAL   @,1X,I10,8X,4(G13.6,3X)             ANACOVA
     M          ,/,15X,@  SLOPES     @,/, 15X,                          ANACOVA
     N 93(1H-),/,15X,@REGRESSION     @,1X,I10,8X,4(G13.6,3X)             ANACOVA
     O          ,/,15X,@  MODEL      @                                  ANACOVA
     P          ,/,15X,@ERROR        @,1X,I10,8X,2(G13.6,3X),/,         ANACOVA
     * 15X,                                                             ANACOVA
     Q 93(1H-),/,15X,@TOTAL          @,1X,I10,8X,  G13.6,/,             ANACOVA
     * 15X,                                                             ANACOVA
     R 93(1H-))                                                         ANACOVA
      WRITE(6,150)                                                      ANACOVA
      IF(IOPT.EQ.1) RETURN                                              ANACOVA
   80 NDF(7)=NDF(7)-1                                                   ANACOVA
      NDF(10)=NDF(10)-1                                                 ANACOVA
      NDF(12)=NDF(12)-1                                                 ANACOVA
      SS(7)=SS(7)-SSU                                                   ANACOVA
      SS(10)=SS(10)-SSU                                                 ANACOVA
      SS(12)=SS(12)-SSU                                                 ANACOVA
      XMS(7)=XMS(7)/NDF(7)                                              ANACOVA
      XMS(10)=XMS(10)/NDF(10)                                           ANACOVA
      F(7)=XMS(7)/XMS(11)                                               ANACOVA
      F(10)=XMS(10)/XMS(11)                                             ANACOVA
      PROBF(7)=FISH(F(7),NDF(7),NDF(11))                                ANACOVA
      PROBF(10)=FISH(F(10),NDF(10),NDF(11))                             ANACOVA
      WRITE(6,101)      (NDF(I),SS(I),XMS(I),F(I),PROBF(I),I=1, 5)      ANACOVA
  101 FORMAT(//,15X,93(1H-),/,38X,@ANALYSIS OF COVARIANCE TABLE (MEAN DIANACOVA
     AFFERENCE)@,/,15X,                                                 ANACOVA
     193(1H-),/,15X,@SOURCE OF@,9X,@DEGREES OF@,5X,@SUM OF@,10X,@MEAN@, ANACOVA
     212X,@F-@,14X,@PROB@,/,15X,@VARIATION@,9X,@FREEDOM@,8X,@SQUARES@,  ANACOVA
     39X,@SQUARE@,10X,@RATIO@,11X,@F@,/,15X,93(1H-),/,                  ANACOVA
     *                15X,@TREATMENTS@,4X,I10,8X,4(G13.6,3X),/,15X,      ANACOVA
     493(1H-),/,                                                        ANACOVA
     A           15X,@1 POPULATION @,1X,I10,8X,4(G13.6,3X)              ANACOVA
     B          ,/,15X,@1 POP.- SLOPE@,1X,I10,8X,4(G13.6,3X)            ANACOVA
     C          ,/,15X,@    OF MEANS @,/,   15X,                        ANACOVA
     * 93(1H-),/,15X,@SLOPE OF MEANS@,I10,8X,4(G13.6,3X)                ANACOVA
     D          ,/,15X,@DIFF MEAN SL.@,1X,I10,8X,4(G13.6,3X)            ANACOVA
     E          ,/,15X,@AND  PAR. SL.@                                  ANACOVA
     F          )                                                       ANACOVA
      WRITE(6,102)      (NDF(I),SS(I),XMS(I),F(I),PROBF(I),I=6,10),      ANACOVA
     2              NDF(11),SS(11),XMS(11),NDF(12),SS(12)                ANACOVA
      WRITE(6,150)                                                      ANACOVA
  150 FORMAT(1H1)                                                       ANACOVA
  999 RETURN                                                            ANACOVA
   99 IOPT=-1                                                           ANACOVA
      RETURN                                                            ANACOVA
      END                                                               ANACOVA
      SUBROUTINE FOURG (DATA,N,ISIGN,WORK)                              FOURG
C     COOLEY-TUKEY FAST FOURIER TRANSFORM IN USASI BASIC FORTRAN.       FOURG
C     ONE-DIMENSIONAL TRANSFORM OF COMPLEX DATA, ARBITRARY NUMBER OF    FOURG
C     POINTS.  N POINTS CAN BE TRANSFORMED IN TIME PROPORTIONAL TO      FOURG
C     N*LOG(N) (FOR N NON-PRIME), WHEREAS OTHER METHODS TAKE N**2 TIME. FOURG
```

```
C       FURTHERMORE, BECAUSE FEWER ARITHMETIC OPERATIONS ARE PERFORMED,   FOURG
C       LESS ERROR IS BUILT UP.  THE TRANSFORM DONE IS--                  FOURG
C       DIMENSION DATA(N),TRANSFORM(N),WORK(N)                            FOURG
C       COMPLEX DATA,TRANSFORM,WORK                                       FOURG
C       TRANSFORM(K) = SUM(DATA(J)*EXP(ISIGN*2*PI*I*(J-1)*(K-1)/N)),      FOURG
C       SUMMED FROM J = 1 TO N FOR ALL K FROM 1 TO N.  THE TRANSFORM      FOURG
C       VALUES ARE RETURNED TO DATA, REPLACING THE INPUT.  N MAY BE ANY   FOURG
C       POSITIVE NUMBER, BUT IT SHOULD BE NON-PRIME FOR SPEED.  ISIGN =   FOURG
C       +1 OR -1.  A -1 TRANSFORM FOLLOWED BY A +1 ONE (OR VICE VERSA)    FOURG
C       RETURNS N TIMES THE ORIGINAL DATA.  WORK IS A ONE-DIMENSIONAL     FOURG
C       COMPLEX ARRAY OF LENGTH N USED FOR WORKING STORAGE.               FOURG
C       RUNNING TIME IS PROPORTIONAL TO N * (SUM OF THE PRIME FACTORS OF  FOURG
C       N).  FOR EXAMPLE, N = 1960, TIME IS TO * 1960 * (2+2+2+5+7+7).    FOURG
C       NAIVE METHODS DIRECTLY IMPLEMENTING THE SUMMATION RUN IN TIME     FOURG
C       PROPORTIONAL TO N**2.  AN UPPER BOUND FOR THE RMS RELATIVE ERROR  FOURG
C       IS 3 * 2**(-B) * SUM(F**1.5), WHERE B IS THE NUMBER OF BITS IN    FOURG
C       THE FLOATING POINT FRACTION AND THE SUM IS OVER THE PRIME         FOURG
C       FACTORS OF N.  WRITTEN BY NORMAN BRENNER, MIT LINCOLN LABORATORY, FOURG
C       AUGUST 1968.  SEE--IEEE TRANSACTIONS ON AUDIO AND ELECTROACOUSTICSFOURG
C       (JUNE 1967), SPECIAL ISSUE ON THE FAST FOURIER TRANSFORM.         FOURG
        DIMENSION DATA(1), WORK(1), IFACT(32)                             FOURG
        TWOPI=6.28318507*FLOAT(ISIGN)                                     FOURG
C       FACTOR N INTO ITS PRIME FACTORS, NFACT IN NUMBER.  FOR EXAMPLE,   FOURG
C       FOR N = 1960, NFACT = 6 AND IFACT(IF) = 2, 2, 2, 5, 7 AND 7.      FOURG
        IF=0                                                              FOURG
        NPART=N                                                           FOURG
        DO 50 ID=1,N,2                                                    FOURG
        IDIV=ID                                                           FOURG
        IF (ID-1) 10,10,20                                                FOURG
   10   IDIV=2                                                            FOURG
   20   IQUOT=NPART/IDIV                                                  FOURG
        IF (NPART-IDIV*IQUOT) 40,30,40                                    FOURG
   30   IF=IF+1                                                           FOURG
        IFACT(IF)=IDIV                                                    FOURG
        NPART=IQUOT                                                       FOURG
        GO TO 20                                                          FOURG
   40   IF (IQUOT-IDIV) 60,60,50                                          FOURG
   50   CONTINUE                                                          FOURG
   60   IF (NPART-1) 80,80,70                                             FOURG
   70   IF=IF+1                                                           FOURG
        IFACT(IF)=NPART                                                   FOURG
   80   NFACT=IF                                                          FOURG
C       SHUFFLE THE DATA ARRAY BY REVERSING THE DIGITS OF THE INDEX.      FOURG
C       REPLACE DATA(I) BY DATA(IREV) FOR ALL I FROM 1 TO N.  IREV-1 IS   FOURG
C       THE INTEGER WHOSE DIGIT REPRESENTATION IN THE MULTI-RADIX         FOURG
C       NOTATION OF FACTORS IFACT(IF) IS THE REVERSE OF THE REPRESENTATIONFOURG
C       OF I-1.  FOR EXAMPLE, IF ALL IFACT(IF) = 2, THEN FOR I-1 = 11001, FOURG
C       IREV-1 = 10011.  A WORK ARRAY OF LENGTH N IS NEEDED.              FOURG
        IP0=2                                                             FOURG
        IP3=IP0*N                                                         FOURG
        IWORK=1                                                           FOURG
        I3REV=1                                                           FOURG
        DO 110 I3=1,IP3,IP0                                               FOURG
        WORK(IWORK)=DATA(I3REV)                                           FOURG
        WORK(IWORK+1)=DATA(I3REV+1)                                       FOURG
        IP2=IP3                                                           FOURG
        DO 100 IF=1,NFACT                                                 FOURG
        IP1=IP2/IFACT(IF)                                                 FOURG
        I3REV=I3REV+IP1                                                   FOURG
        IF (I3REV-IP2) 110,110,90                                         FOURG
```

```
 90    I3REV=I3REV-IP2                                                  FOURG
 100   IP2=IP1                                                          FOURG
 110   IWORK=IWORK+IP0                                                  FOURG
       IWORK=1                                                          FOURG
       DO 120 I3=1,IP3,IP0                                              FOURG
       DATA(I3)=WORK(IWORK)                                             FOURG
       DATA(I3+1)=WORK(IWORK+1)                                         FOURG
 120   IWORK=IWORK+IP0                                                  FOURG
C      PHASE-SHIFTED FOURIER TRANSFORM OF LENGTH IFACT(IF).             FOURG
C      IPROD=IP1/IP0                                                    FOURG
C      IREM=N/(IFACT(IF)*IPROD)                                         FOURG
C      DIMENSION DATA(IPROD,IFACT(IF),IREM),WORK(IFACT(IF))             FOURG
C      COMPLEX DATA,WORK                                                FOURG
C      DATA(I1,J2,I3) = SUM(DATA(I1,I2,I3) * W**(I2-1)), SUMMED OVER    FOURG
C      I2 = 1 TO IFACT(IF) FOR ALL I1 FROM 1 TO IPROD, J2 FROM 1 TO     FOURG
C      IFACT(IF) AND I3 FROM 1 TO IREM.                                 FOURG
C      W = EXP(ISIGN*2*PI*I*(I1-1+IPROD*(J2-1))/(IPROD*IFACT(IF))).     FOURG
       IF=0                                                             FOURG
       IP1=IP0                                                          FOURG
 130   IF (IP1-IP3) 140,240,240                                         FOURG
 140   IF=IF+1                                                          FOURG
       IFCUR=IFACT(IF)                                                  FOURG
       IP2=IP1*IFCUR                                                    FOURG
       THETA=TWOPI/FLOAT(IFCUR)                                         FOURG
       SINTH=SIN(THETA/2.)                                              FOURG
       ROOTR=-2.*SINTH*SINTH                                            FOURG
C      COS(THETA)-1, FOR ACCURACY                                       FOURG
       ROOTI=SIN(THETA)                                                 FOURG
       THETA=TWOPI/FLOAT(IP2/IP0)                                       FOURG
       SINTH=SIN(THETA/2.)                                              FOURG
       WSTPR=-2.*SINTH*SINTH                                            FOURG
       WSTPI=SIN(THETA)                                                 FOURG
       WMINR=1.                                                         FOURG
       WMINI=0.                                                         FOURG
       DO 230 I1=1,IP1,IP0                                              FOURG
       IF (IFCUR-2) 150,150,170                                         FOURG
 150   DO 160 I3=I1,IP3,IP2                                            FOURG
       J0=I3                                                            FOURG
       J1=I3+IP1                                                        FOURG
       TEMPR=WMINR*DATA(J1)-WMINI*DATA(J1+1)                            FOURG
       TEMPI=WMINR*DATA(J1+1)+WMINI*DATA(J1)                            FOURG
       DATA(J1)=DATA(J0)-TEMPR                                          FOURG
       DATA(J1+1)=DATA(J0+1)-TEMPI                                      FOURG
       DATA(J0)=DATA(J0)+TEMPR                                          FOURG
 160   DATA(J0+1)=DATA(J0+1)+TEMPI                                      FOURG
       GO TO 220                                                        FOURG
 170   IWMAX=IP0*IFCUR                                                  FOURG
       DO 210 I3=I1,IP3,IP2                                             FOURG
       I2MAX=I3+IP2-IP1                                                 FOURG
       WR=WMINR                                                         FOURG
       WI=WMINI                                                         FOURG
       DO 200 IWORK=1,IWMAX,IP0                                         FOURG
       I2=I2MAX                                                         FOURG
       SUMR=DATA(I2)                                                    FOURG
       SUMI=DATA(I2+1)                                                  FOURG
 180   I2=I2-IP1                                                        FOURG
       TEMPR=SUMR                                                       FOURG
       SUMR=WR*SUMR-WI*SUMI+DATA(I2)                                    FOURG
       SUMI=WR*SUMI+WI*TEMPR+DATA(I2+1)                                 FOURG
       IF (I2-I3) 190,190,180                                           FOURG
```

```
 190  WORK(IWORK)=SUMR                                                      FOURG
      WORK(IWORK+1)=SUMI                                                    FOURG
      TEMPR=WR                                                              FOURG
      WR=WR*ROOTR-WI*ROOTI+WR                                               FOURG
 200  WI=TEMPR*ROOTI+WI*ROOTR+WI                                            FOURG
      IWORK=1                                                               FOURG
      DO 210 I2=I3,I2MAX,IP1                                                FOURG
      DATA(I2)=WORK(IWORK)                                                  FOURG
      DATA(I2+1)=WORK(IWORK+1)                                              FOURG
 210  IWORK=IWORK+IP0                                                       FOURG
 220  TEMPR=WMINR                                                           FOURG
      WMINR=WMINR*WSTPR-WMINI*WSTPI+WMINR                                   FOURG
 230  WMINI=TEMPR*WSTPI+WMINI*WSTPR+WMINI                                   FOURG
      IP1=IP2                                                               FOURG
      GO TO 130                                                             FOURG
 240  RETURN                                                                FOURG
      END                                                                   FOURG
      SUBROUTINE LAGCOR(LA,A,LB,B,C,LSTART,LSTOP)                           LAGCOR
C                                                                           LAGCOR
C     THIS ROUTINE CALCULATES A SAMPLE CROSS-CORRELATION OF THE RECORD      LAGCOR
C          A OVER THE RECORD B WITH LAGS BETWEEN LSTART AND LSTOP AND       LAGCOR
C          STORES THE RESULT IN C                                           LAGCOR
C    **** CAUTION ***** THERE IS NO CHECK FOR A ZERO RECORD                 LAGCOR
C                                                                           LAGCOR
      DIMENSION A(LA),B(LB),C(LA)                                           LAGCOR
      DO 50 J=LSTART,LSTOP                                                  LAGCOR
      U=0.0                                                                 LAGCOR
      SUMA=0.0                                                              LAGCOR
      SUMB=0.0                                                              LAGCOR
      SA=0.0                                                                LAGCOR
      SB=0.0                                                                LAGCOR
      IF(LB-(LA-J+1)) 10,10,20                                              LAGCOR
   10 N=LB                                                                  LAGCOR
      GO TO 30                                                              LAGCOR
   20 N=LA-J+1                                                              LAGCOR
      IF(N.GT.0) GO TO 30                                                   LAGCOR
      DO 25 I=J,LSTOP                                                       LAGCOR
   25 C(I)=-2.                                                              LAGCOR
      RETURN                                                                LAGCOR
   30 EN=N                                                                  LAGCOR
      DO 40 I=1,N                                                           LAGCOR
      IJ=I+J-1                                                              LAGCOR
      SUMA=SUMA+A(IJ)                                                       LAGCOR
      SUMB=SUMB+B(I)                                                        LAGCOR
      SA=SA+A(IJ)*A(IJ)                                                     LAGCOR
      SB=SB+B(I)*B(I)                                                       LAGCOR
   40 U=U+A(IJ)*B(I)                                                        LAGCOR
      SUMA=SUMA/EN                                                          LAGCOR
      SUMB=SUMB/EN                                                          LAGCOR
      SA=SA-SUMA*SUMA*EN                                                    LAGCOR
      SB=SB-SUMB*SUMB*EN                                                    LAGCOR
   50 C(J)=(U-EN*SUMA*SUMB)/SQRT(SA*SB)                                     LAGCOR
      RETURN                                                                LAGCOR
      END                                                                   LAGCOR
```

APPENDIX E - Format of Retrieval Cards

APPENDIX E - FORMAT OF CARDS REQUIRED FOR RETRIEVAL

E.1 General Structure

All input retrieval cards have the following format:

FORMAT(A4,1X,A4,1X,14I5)

Columns 1 - 4 Contain the key word for a control card.

Column 5 Is ignored and may contain the last character of a key word.

Columns 6 - 10 Contain a qualifier word. For a JOB card this word must be DATE, VOLU or BOTH. For other cards this word must be EACH, ALL, RANGE or blanks. A blank field is equivalent to ALL.

The following is a list of the valid key words:

JOB - Contains JOB ID, date save option, total volume save option, and plot control.

STUDY - Specifies which studies are to be retrieved.

DATE - Specifies which dates are to be retrieved.

MAN - Provision to specify man to be retrieved, currently not implemented as program always uses man 1, 2 and 3.

TYPE - Specifies which sample types are to be retrieved.

TEST - Specifies which tests are to be retrieved.

END - End of run.

EOF - End of job.

E.2 Specific Retrieval Cards

JOB CARD

Column 1 - JOB

The second field of the JOB card (columns 6-10) must contain one of the following parameters. The parameter must always begin in column 6.

DATE - Indicates that the Julian date of the samples being retrieved is to be saved for output. Always required for plot and most analysis.

VOLUM - Indicates that the total 24-hour urine volume for the dates being retrieved is to be saved for output.

BOTH - Indicates that both dates and total volumes are to be saved for output.

(BLANK) - A blank field or mispelled words will cause an error message and the default DATE to be used.

The third field of the JOB card (columns 11 - 15) provides a place for the user's job identification number. It may be any 5 digit integer. However, if the integer is less than 1000, no plots will be produced.

The remaining thirteen 5 column fields of the JOB card are for use by the statistical portion of this program. Currently these are not used, but in the future they can be used for control of statistical subroutines. They must always be 5 digit integers.

STUDY CARD

COLUMN	1	STUDY
COLUMNS 6 - 10		RANGE, EACH, ALL or BLANK
		Blank or ALL - retrieves data from all missions
		EACH - retrieves data from study numbers specified in Col. 15, 20, 25, 30, 35, 40, 45...60.
		RANGE - Study range from Col. 15 to Col. 20.

Study codes are found in Table E.3.

DATE CARD

COLUMN	1	DATE
COLUMNS 6 - 10		EACH, ALL, Blank Range
	ALL = Blank	Get all dates
	EACH	Use individual dates in Col. 15 → up.
	RANGE	Date range from Col. 15 to Col. 20

Dates are Julian Dates. Date ranges for specific missions are found in Table E.3.

MAN CARD

COLUMN	1	MAN
COLUMNS 6 - 10		Same as DATE card except Man numbers are specified. (See Table E.3.)

TYPE CARD

COLUMN	1	TYPE
COLUMNS 6 - 10		Type codes are same as DATE card, found in Table E.3.

TEST CARD

COLUMN	1	TEST
COLUMN	6	RANGE
COLUMNS 11 - 15		Starting test of range
COLUMNS 16 - 20		Ending test of range

If only one test, the test must appear in both fields. See Table B.1 for valid test numbers.

END CARD

When an END card is encountered, the program starts the retrieval. Therefore, all search criteria indicated on the JOB, STUDY, DATE, TYPE & TEST cards must be read in before the END card. There must be an END card for each run.

The first run must contain at least one JOB card, STUDY card, DATE card, TYPE card, and TEST card. These initial parameters remain as the search criteria for each succeeding run, unless changed by another JOB, STUDY, DATE, TYPE or TEST card. There may be no more than one JOB card per run, but each run may have as many STUDY, DATE, TYPE & TEST cards as required.

EOF CARD

When an EOF card is encountered, the program stops and the total job is considered finished. There may be as many runs within a job as needed by the user. Several jobs may be stacked. Parameters set in one job remain constant unless changed.

TABLE E.3

NUMBER CODES FOR DATA FORMS

STUDY	TYPE	DATA FORMS
1 = SMEAT	1 = FRESH URINE	-1 = TEST NOT PERFORMED
2 = SKYLAB 2	2 = FROZEN URINE	-2 = LESS THAN DETERMINABLE
3 = SKYLAB 3	3 = LYOPHILIZED URINE	-3 = TO FOLLOW
4 = SKYLAB 4	4 = URINE SKYLAB BAG SAMPLES (PREFLIGHT & INFLIGHT)	-4 = CALCULATED AS COMBINED VALUES
5 = APOLLO 17	5 = PLASMA	-5 = LESS THAN A 24 HR. PERIOD
6 =	6 = SPECIAL PLASMA - PREFLIGHT LYTHEIUM EDTA	
7 =	7 = SERUM	
8 =	8 = 10% URINE SAMPLE	
9 =	9 = FROZEN PLASMA	
	10 =	
	11 =	
	12 =	
	13 =	
	14 =	
	15 =	

JULIAN DATES

	START	LAUNCH	SPLASH DOWN	END
SMEAT	180	208	265	
SKYLAB 2	114	144	172	
SKYLAB 3	189	209	268	285
SKYLAB 4	283	320	404*	
APOLLO 17	290	342	354	

*Date is 39 of following year

MAN	SMEAT	APOLLO 17	SKYLAB 2	SKYLAB 3	SKYLAB 4
1 = CDR	Crippen	Cernan	Conrad		
2 = PLT	Bobko	Evans	Weitz		
3 = SPT	Thornton	Schmitt	Kerwin		
4 = C1	Alexander	Duke	Schweikart		
5 = C2	Strzynek	Roosa	Musgrave		
6 = C3	Laird	Young	Mc Candless		
7 = C4	Kimzey	Alexander	Alexander		
8 =		Buchanan	Hordinsky		
9 =		La Pinta	La Pinta		

E.4 Sample Input Card Decks

```
JOB DATE      400
STUDY EACH      3
DATE ALL
TYPE EACH       1
TEST RANGE    519 523
END
EOF
```

```
JOB DATE     4300
STUDY EACH      3
DATE ALL
TYPE EACH       5
TEST RANGE    312 314
END
JOB DATE     4301
TEST RANGE    317 320
END
EOF
```

APPENDIX F - MISCELLANEOUS INFORMATION

F.1 - SAMPLE NUMBERS

Sample numbers as they exist in the sample directory for the various missions:

SMEAT	2-674
APOLLO 17	675-936

Sample number for the Skylab missions are not separated.

SKYLAB 2	937-1490
SKYLAB 3	937-2256
SKYLAB 4	937-2256

F.2 ADDITION OF NEW DATA ITEMS

Program Build

1. Increase Dimensions On:

   INDX
   INDX1
   INDX2
   INDX3
   INDX4
   INDX5

   The order of additions is determined by the form number and form position of the new item.

2. In data array of INDX add in test number of new data.

3. Data array of INDX1 add in a U or B for new data in position corresponding to position in INDX.

4. Add appropriate name into INDX2 array.

5. Add index of item in mass storage record into INDX3.

6. Add form number into INDX4.

7. Add field number on card form.

8. Decode statement and format for card number involved changed increase IEND and DO parameter.

9. Add 1 to all following K values. These are start positions in mass storage record. Note DIET parameters work down from top.

10. Increase end parameter for DO 81 in UPDATE card processor.

11. Mass storage records must be checked to insure space.

Program Retrieve

1. In subroutine RETRIEVE

   LREC is maximum 125 for URINE and 25 for BLOOD - if additions run over these numbers dimensions must be changed on LTEST, IBL and/or IUR.

2. In subroutine TEXT

Insert TEXT name of new item into MBLD or MURN array indexed by test #. May require additional space.
Insert units as above into MITS, MITB.

K	IEND	
2	7	2 8
8	12	9 12
19	16	20
34	10	35
43	5	44
47	16	48
62	15	63
76	16	77
91	15	92
116	11	117

Search INDX4 for card number

Check INDX5 for match to J else continue search

Pickup INDX3 as index for store

F.3 AUXILLIARY FILES

1. All data is for mission 3 (SKYLAB 3)
2. Each file is for one man
3. Data is from URINE file including DIET Data
4. File names FMAN1, FMAN2, FMAN3 for CDR, PLT, and SPT, resp.
5. FORTRAN Unformatted 130 words/record

Word		
1 and 126	Sample number	Integer
2-105	Urine Sample Data	Real (See index Table B.1)
106-115	Not Used	
116-125	DIET Data	Real
126	Sample number	Integer
127	STUDY	All study 3
128	JULIAN DATE	
129	MAN	All 1 in file FMAN1, 2 in FMAN2, etc.
130	Sample Type	All URINE and DIET is type 1

F.4 AVAILABLE FILES AND CARD DECKS

BUILD

This program exists as a MODIFY program library (BUILD PL) and also as a card deck. A run deck to cause BUILD to create new files is labelled CREATE and is listed in Section 3. A run deck to cause BUILD to update existing files is labelled UPDATE and is also listed in Section 4. (See Appendix D for listings.)

TYPLOK

This program exists only as a card deck and is a programmers tool to insure integrity of the file structure. (See Appendix D for listings.)

RETRIEVAL

This is the primary retrieval and analysis program, it exists as a MODIFY program library. A deck of control cards exists to cause execution of the program. Also sample data cards are available. (See Appendix D for listings.)

F.5 SAMPLE DIRECTORY RECORD LAYOUT: FILE RA1 CONTENTS (00 C RDS)

Record Length 35 words

Words 1-12 and 14 are binary integer. Word 13 is binary real.

Refer to Appendix E for variable values

Word	Variable
1	Sample Number
2	Study Number (Mission Code)
3	Julian Date of Sample
4	Man Number
5	Sample Type
6	GMT
7	Month of Sample
8	Day of Sample
9	Year of Sample
10	TOTAL Volume ml
11	BAG Volume ml
12	BAG Number
13	VDR
14	Master Sample Number
15-25	Alpha Remarks
26-35	Unused