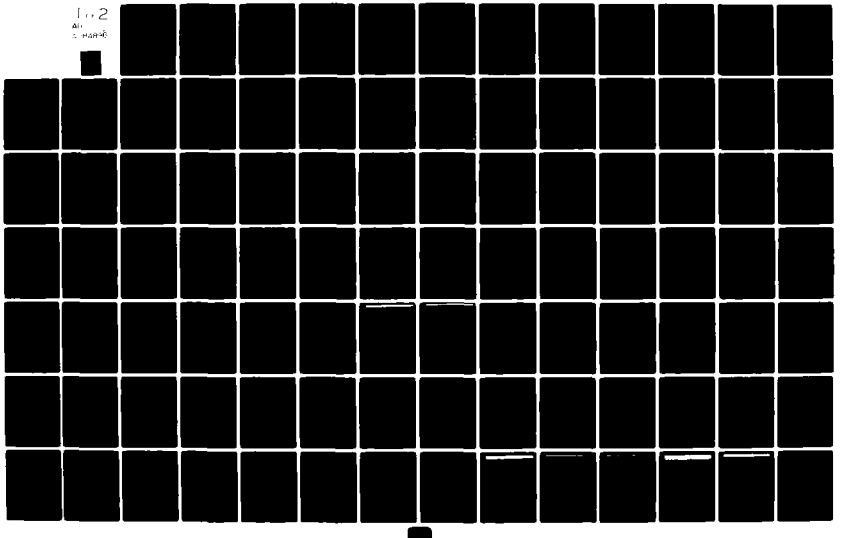
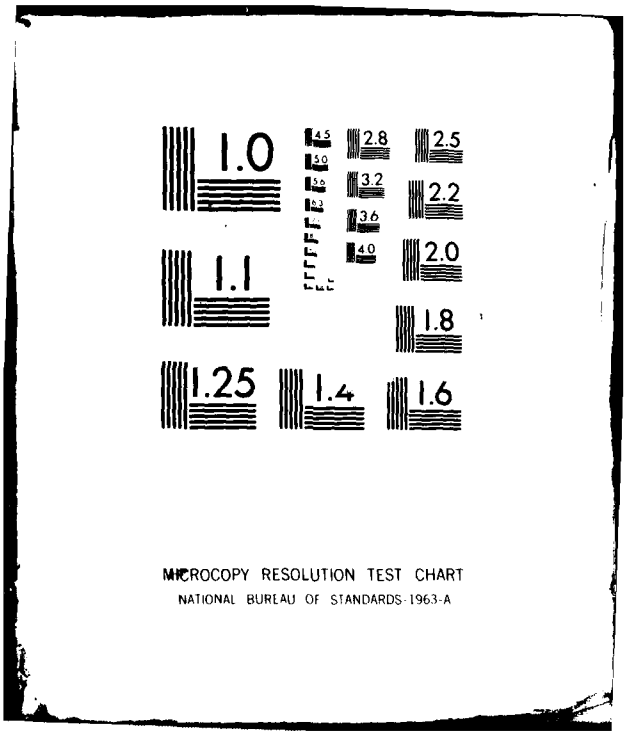
AD-A084 896 BATTELLE COLUMBUS LABS OHIO TACTICAL TECHNOLOGY CENTER F/G 9/2
A QUALITY CONTROL DATA MANAGEMENT SYSTEM.(U)
FEB 78 M B NEHER, R E HEFFELFINGER DAAK40-73-C-0142
UNCLASSIFIED BATT-CDIR-1-2 DRXTH -TD-CR-80069 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) Unclassified

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: DRXTH-TD-CR-80069	2. GOVT ACCESSION NO.: AD-A084896
3. RECIPIENT'S CATALOG NUMBER:	
4. TITLE (and Subtitle): A Quality Control Data Management System	5. TYPE OF REPORT & PERIOD COVERED: Final June 77- Feb 78
6. PERFORMING ORG. REPORT NUMBER: BATT-CDIR-1-2	
7. AUTHOR(s): M.B. Neher, R.E. Heffelfinger	8. CONTRACT OR GRANT NUMBER(s): DAAK40-73-C-0142
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Battelle Columbus Laboratories, 505 King Avenue, Columbus, OH 43201	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 822896.R 111
11. CONTROLLING OFFICE NAME AND ADDRESS: US Army Toxic and Hazardous Materials Agency, Aberdeen Proving Ground, Maryland 21010	12. REPORT DATE: 28 Feb 78
	13. NUMBER OF PAGES: 100
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

LEVEL II

16. DISTRIBUTION STATEMENT (of this Report)

Cleared for public release; distribution unlimited.

Final rept. Jun 77-Feb 78

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

DTIC ELECTE MAY 28 1980 S C D

18. SUPPLEMENTARY NOTES

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

Laboratory Quality Control
Data
Installation Restoration

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

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

A laboratory quality control system has been programmed for the Hewlett-Packard 3384 Computer used in Installation Restoration analytical chemistry laboratories.

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

Unclassified 140052
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AD A084896

DDC FILE COPY

DISCLAIMER NOTICE

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

REPORT CDIR 1-2

A QUALITY CONTROL DATA MANAGEMENT SYSTEM

M. B. NEHER AND R. E. HEFFELFINGER
BATTELLE
COLUMBUS LABORATORIES
TACTICAL TECHNOLOGY CENTER
505 KING AVENUE
COLUMBUS, OHIO 43201

28 FEBRUARY 1978
WORKING REPORT

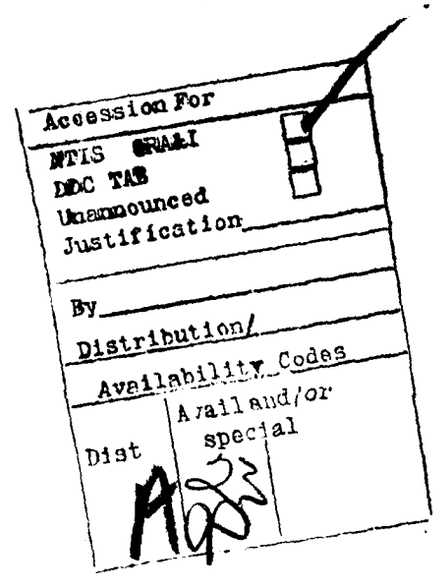
SPONSORED BY

PROGRAM MANAGER, CHEMICAL DEMILITARIZATION
AND INSTALLATION RESTORATION
ABERDEEN PROVING GROUND, MARYLAND 21010
MIPR NOS. 7.70023, 7.6P510, AND 7.70047

U. S. ARMY MISSSILE RESEARCH & DEVELOPMENT COMMAND
REDSTONE ARSENAL, ALABAMA 35809

PREFACE

This research was supported by the Program Manager, Chemical Demilitarization and Installation Restoration, Aberdeen Proving Ground, Maryland, and was monitored by the U. S. Army Missile Research and Development Command under Contract No. DAAK40-73-C-0142 sponsored by the Defense Advanced Research Projects Agency.



DISCLAIMER

The views and conclusions contained in this document are those of the authors and should not necessarily be interpreted as representing the official policies, either expressed or implied, of the Program Manager, Chemical Demilitarization and Installation Restoration, Aberdeen Proving Ground, Maryland, or of the U. S. Government.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

A QUALITY CONTROL DATA MANAGEMENT SYSTEM

by

M. B. Neher and R. E. Heffelfinger

SUMMARY

A data management system has been developed for the HP 3354 data system to support a quality control program. This system assigns laboratory numbers to incoming samples and keeps track of the status of each sample as results are entered by the various analysts. It also assigns quality control sample designators (blanks, duplicates, and standards). These data from the standard samples are used to calculate precision, accuracy, and detection limit. Blanks, duplicates, and standards are used to keep a cumulative sum (CuSum) file for both precision (duplicates), and accuracy (blanks and standards). The CuSum files are used to determine whether or not an analysis is "in control".

When samples have been completed, the data will be transmitted to Edgewood Arsenal for permanent storage.

DATA MANAGEMENT SYSTEM ORGANIZATION

This system consists of several sets of programs which access and manipulate data using several sets of data files. These programs and data files are described in this section.

System Concept

This system is designed to be used on a HP 3354 GC Data System, operating in terminal mode Basic. Each sample is catalogued as separate files in the systems directory, with the required sample identification data in a header record and each analyte in a separate logical record. Both user samples and Q.C. samples are catalogued, but under different formats. Q.C. samples (duplicates, blanks, and standards) are employed to determine the precision, accuracy, and detection limit for each analyte

in each sample type. The detection limit is determined by the method of Hubeaux and Voss[a]. Data for standards are recorded in a statistical data base file, which contains partial sums for each concentration. This data allows calculation of the detection limit, accuracy and precision as a function of concentration and variance for both precision and accuracy. The data base is updated with each set of standards, applying a weighing factor to the data already in the data base.

The variance determined from this data base is used to set up cumulative sum data files.[b] All data needed to evaluate the performance of the system are contained in the statistical data base and cumulative sum data files. As samples are logged into the system, blank, duplicate, and standard samples are specified by the system based on the number of samples. The order is determined by a random number system. Basic does not run creation of files from within a program. Hence, samples are entered into temporary files and then written into permanent files after the operator has created the needed files. Analysts interact with the system in two ways, (1) to notify the system that a given analysis is being started, and (2) to enter the results of the analysis.

The system maintains a cross reference file which contains the Users Sample Number and the Laboratory Number which was assigned to the sample. The cross reference file can be queried to determine the laboratory number for a sample, and the status file is then queried to determine how many analyses remain on the sample.

When an analysis is found to be out of control (or approaching the out-of-control point), the cusum files can be plotted on one of the system terminals to determine what mode of failure has occurred. Utility programs are provided for restarting when such failures occur.

---

[a] Hubaux, A., and Voss, G., "Precision and Detection Limits for Linear Calibration Curves", Anal. Chem., 42, 849 (1967).

[b] Anon, "Laboratory Quality Control Manual", Federal Water Pollution Control Administration, Robert S. Kerr Water Research Center (1969).

Programs

The programs used in the Q.C. Data Management System can be grouped together conveniently into several categories: sample entry, data entry, initial start-up and utility programs. These groupings are discussed in the following sections.

Sample Entry Program

Samples are logged into the system by this program, which consists of five sub-programs. Four of these chain together and produce an intermediate file of data for use by the fifth, and a list of files to be created by the operator, along with the identification of each file, i.e., users sample number, levels and identification of QC samples (i.e., duplicates, standards, blanks, and spikes). The last program is called when these samples are ready to be put in the process stream, and enters them into the system.

These sub-programs and their functions are as follows:

ENTERS: SV - Dialogue to enter and set up parameters for samples. Sets up random sequences for grouped samples. File size limits the sequence length to ∿50, so the maximum group size is 25 (tentatively). For non-grouped samples, the maximum group size of 25 is used and repeated as needed, selecting a new random sequence each time. It chains to ENTRS2:SV for sample information input. If the sample is a "spectral" (i.e., non-standard) sample, it chains to ENTRSP:SV.

ENTRSP:SV - Special samples are handled by this sub-program. A temporary file of information for the last segment is set up (SPCTEM:DA) containing the sample identification and analyses for each sample. (Not yet implemented.)

ENTRS2:SV - Accepts sample information for regular samples and sets up temporary file of information (TEMFIL:DA) which contains sample identification, type of analysis, and specification of Q.C. samples. This sub-program also prepares a list as the terminal of Lab Sample Numbers, corresponding User Sample Numbers and file lengths required. If a Q.C. sample is required, the concentrations and type of Q.C. sample are printed out rather than the User Sample Numbers..

FILWRT:SV - This program is run after the samples have been prepared for analysis (i.e., Lab Numbers put on bottles and Q.C. samples prepared). It puts entries in the status file (STATUS:DA) and the cross reference file (CREFT:DA).

Variables are communicated between segments through and retained during chaining by common arrays G, B, F, P, C, and L.

B - contains the sequence identifiers set up by ENTRS1:SV.

G - contains the parameters passed between sub-programs:

G(1) = Group size (samples, excluding Q.C.)

G(2) = Identifies group type: -1 = ungrouped sample
0 = new group
1 = grouped sample

G(3) = Sample type (read from QCTIME:DA)

G(4) = Group type (read from QCTIME:DA) (0 = unset)

G(5) = Number of user samples remaining in sequence

G(6) = Pointer to position in sequence

G(7) = Next standard concentration

G(8) = Last location in present sequence.

F - contains current sample and last started sample.

P - contians pointers into TEMFIL:DA or SPCTEM:DA.

C - contains prespecified standards ratios.

L - contains record length information read from METHOD:DA.

Data Entry Program

This program is the program by which analysis data is entered into the system. Two entry modes are provided, normal entry (at a terminal) for systems which are not automated, and automatic entry for instruments (currently GC's only) which are automated and produce a computer file containing data. It consists of form sub-programs which chain together.

These sub-programs and their functions are as follows:

ENTERD:SV Dialogue enter sample number, compound name, date, analysis method, analysts initials, and instrument number. For Q.C. samples, it chains to ENTRD2:SV. Replicates are handled by ENTRS1. For user samples, it checks the Cusum files for accuracy and precision. If the system is "out of control", a message is printed and the data is not entered in the file. If the system is in control, but approaching the "out of control" condition, the data is entered in the file and a warning message is printed at the terminal. ENTERD chains to ENTRD4 for the actual file entry and STATUS file updates.

ENTRD1:SV This sub-program handles duplicates. The first of a pair of duplicates is a user sample. Data are entered in this file in regular format, and also in the duplicate (Q.C.) file in the "expected" value location. It chains to ENTRD2:SV.

ENTRD2 - This sub-program handles data for Q.C. samples (duplicates, standards or spikes, and blanks). Data for the duplicates are used to update the precision CuSum file. Standards, spikes, and blanks data are used to update the accuracy CuSum file. The standard or spike data are stored until a complete set has been accumulated. At this time, it chains to ENTRD3:SV.

ENTRD3 - This sub-program uses data stored in temporary locations to update the data base and calculate new precision, accuracy and detection limit data. These new data are written back on this file for use by ENTRD1.

ENTRD4 - This sub-program uses data packed by ENTRD1 to update the file header, and then update the status file.

Common variables for communicating between ENTERD programs.

COM H$(28) Sample file header
- 1-6 Sample Number
- 7-10 Sample Type
- 11-16 User Sample Number
  - 11-15 are blank if Q.C.
  - 16 = index to std conc.
  - (1=1/2X, 2=X, 3=2X, 4=5X, 5=10X, 6=20X, 7=50X)
- 17-21 Date Submitted
- 22-23 File
- 24 Priority
- 25 Replicate Sequence*
  - 1 = first
  - 2 = second, etc.
- 26 No. Replicates*
  - 0 = single sample
  - 1 = duplicates
  - 2 = triplicates, etc.

*Initial system is restricted to duplicates, of which the second is a Q.C. sample.

27,28	Status
	0 = not started
	>0 = number analyses remaining
	-1 = done
	(if applicable, -2 = all replicates done)
N$(9)	Data File Name
V$(20)	Analysis Data String
1-5	Analyte Name
6-10	Date Completed
11,12	Analysis Method
13,14	Instrument Number
15-17	Analysts Initials
18-20	Measurement Units
P(6)	Numeric Variables
1	Analysis Type
2	Sample File Record Number
3	Address of current string in record
4	Replicate Identifier
5	Analysis Result
6	Analysis Sequence Number in Methods File
7	Number of Results packed in B Array
A(2)	Numeric Equivalent of Sample Number

Initial Start-Up Programs

USTATS:SV

This program accepts data from analysis of standards, without requiring that the standards be logged into the system under the normal laboratory number system (i.e., by use of the sample log in program). It is to be used only for initial start up, or to rebuild the data base

after a method goes out of control. It accepts data in sets run at the standard concentrations (.5, 1, 2, 5, 10, 20, 50 X detection limit) and stores it in the data base as sums at each concentration, ΣX, ΣX^2, ΣY, ΣY^2, $\Sigma X \cdot Y$, $\Sigma \Delta Y$ and $\Sigma(\Delta Y)^2$ and also stores the number of sets of data, a minimum of 4 sets of data. The detection limit can be calculated, using the method of Hubeaux and Voss. For this calculation, the program chains to HUBVOS:SV. The accuracy and precision are also calculated, and the detection limit and the variances and coefficients for the least squares best fit line for precision and accuracy are then added to the data base.

HUBVOS:SV

This sub-program is used for the statistical calculations of detection limit, precision and accuracy.

FXCSUM:SV

Once the data base has been built for and analyzed, this program is run to set up the cusum file for the method. Once the cusum file has been set up, it will automatically be updated as data are entered into the system through ENTERD. Hence, as with USTATS, it's only used for initial creation of the cusum files or to rebuild a file when a method goes out of control.

Maintenance and Utility Programs

This category includes those programs which are used to examine the status of the system.

CREF:SV

This program is used to examine the cross reference table to determine the Laboratory Number corresponding to a given User Number, to look up that Number in the Status Table, and report on its status. If requested, it will then look up the specific Sample File and report the status of each analysis in the sample.

CKCSUM:SV

This program is used to examine the cusum file when a method is reported out of control or heading towards being out of control. It puts a rough plot on the system device (TTY or CRT) or alternately, can send the data to a remote computer (e.g., Tektronix 4051) for output on any desired output medium.

WSTATS:SV

This program is used for initialization of the Status File, STATUS:DA. It enters the prefix for the Laboratory Number (specified by the operator) and sets the value of the laboraotry to 00001, or any other value entered by the operator. It is also used to change the prefix designation.

WMETHD:SV

This program is used to set up the file of methods information used by the various programs in the system, and to rewrite the Method File as changes become needed.

WTMFIL:SV

This program is used to initialize the temporary files for ENTERS and FILWRT. It will be used only to reinitialize these temporary files in the event of an unexpected incident preventing their normal reinitialization by the system programs.

WQCTIM:SV

This program is used to initialize the Q.C. time file, and as with WTMFIL:SV, should not be needed again unless an unexpected malfunction occurs.

GRPSZE:SV

This program is used to create the Group Size Table. This table should never be altered by the system. However, it is expected that changes in the group sizes will be required as data are acquired in the system.

TAB90T:SV

This program is used to write out the T Table File for the statistical program portion of USTATS:SV and ENTERD:SV. It should not be needed again barring a catastrophic system failure.

CRFLST

This program is used to initialize the cross reference table, CREFT:DA. If you run a cross reference file which has been already initialized, it will list the contents of the file. (It is not for use on any other file and would probably give an error message.)

FILCHK

This program was written for debugging purposes, and is included for help if problems arise. It is for use only on sample (user or Q.C.) files and simply lists the header and each analysis block, without spacing or decoding.

Data Files

The programs described above access and manipulate 4 sets of data files:

(1) Sample files - one file for each sample.

(2) Statistical Data Files - two files for each defined analysis type. One file contains data from standard samples for calculation of the precision,

accuracy, and detection limit for each analyte. The second file keeps the cumulative data for accuracy and precision of each analysis.

(3) Status and Cross Reference Files - The Status File has an entry for each sample, arranged in numerical order. The Cross Reference File contains the User Number and the Laboratory Number.

(4) Parameter Files - These files contain parameters which are used by the various programs in recording and manipulating of data and data files.

These files are described more fully in the following section.

Sample Files (ANNNNN:DA)

Header (Real and Q.C. Samples) 19 words		
Lab No. Dec. Eq.	Wd 1, 2, 3, 4	(2 Data words)
Lab Sample No.	Char. 1-6	
Analysis Type	Char. 7-10	
User Sample No.	Char. 11-16	(Note 1)
Julian Date Received	Char. 17-21	
File	Char. 22-23	
Priority	Char. 24	
Number of Replicates	Char. 25	(Note 2)
Replicate Sequence No.	Char. 26	
Status Flag	Char. 27-28	(Note 3)

Each analysis for real samples (21 words for each analyte)	
Analysis Name	Char. 1-5
Julian Analysis Date	Char. 6-10
Analysis Method	Char. 11, 12
Measurement Boolean	Char. 13, 14

Measurement Mantissa	Char. 15-20	
Measurement Exponent	Char. 21-23	
Measurement Units	Char. 24-26	
Accuracy	Char. 27-30	
Precision	Char. 31, 32	
Instrument Number	Char. 33, 34	
Analysts Initials	Char. 35-37	
Status Flag	Char. 38	(Note 4)
Analysis Sequence No.	Char. 39, 40	(Note 5)

QC Sample Files (16 words for each analyte)

Analysis Name	Char. 1-5
Julian Date	Char. 6-10
Method	Char. 11, 12
Instrument No.	Char. 13, 14
Analyst Initials	Char. 15-17
Status Flag	Char. 18
Expected Value	2 words (1 data word)
Found Value	2 words (1 data word)
Analysis Sequence No.	2 words (1 data word)

Note 1: For QC Samples, characters 1-5 are blank
Char. 6: 1 = blank
2-8 are relative levels for standards

Note 2: 0 for single samples, 2 for duplicates (Q.C.)
3-9 indicates spikes.
The second character indicates offset from first replicate or the base sample for the spike.

Note 3: 0 = not started
-1 = Done
-2 = All replicates done (if applicable)
>0 = No. of analyses to be completed.

Note 4: 0 = not started
1 = started
2 = completed

Note 5: N = sequence identifying number in Methods file

REAL SAMPLES: Block 1 Header+ 5 analyses
Subsequent blocks 6 analyses

QC SAMPLES: Block 1 Header+ 6 analyses
Subsequent blocks 8 analyses

Temporary Files

TEMFIL:DA (6 records)

Temporary file to hold list of Laboratory Numbers for use by FILWRT to write unto Analysis File after creation by operator.

Lab No. (N$)	1-6	
User No. (U$)	7-12	22 char. (12 words)
Date (D$)	13-17	
Analysis Type	18	
Priority	19	
No. Replicates	20	
Replicate Sequence	21	
Unused	22	

First block

Word 121, 122	Block number of last entry	P(1)
123, 124	Address of last word of last entry	P(2)
125, 126	Number of entries available	P(3)

SPCTEM:DA (6 blocks)

Temporary file for special samples (tentative format).

Length of entry	1, 2	(holds number of anlayses)
Lab No. (N$)	1-6	10 words as in TEMFIL:DA
User No. (U$)	7-12	
Date (D$)	13-17	
Analysis No. 1	Use sequence	
Analysis No. 2	numbers from	
etc.	Methods File	

Sample Status Files

STATUS:DA Control File

1. Packed decimal equivalent of prefix + 2 digits.
2. 3 low digits of next lab number.
3. Same for highest lab number for
4. which analysis has been started.

eated
5. Same as above
6. for lab number.

h
7. No. of samples in set

ple
set to 0 when done.

8. Sample type initially.

when started:

thousands value - no. analyses started

units value - no. analyses finished

CREFT:DA Cross Reference File

1.	Users Lab No.	6 char.	Repeated for each sample.
2.	Lab No.	6 char.	

Quality Control Data Files

CUSUMN:DA Cusum File (one for each sample type)
Two blocks for each analysis, first block Precision
second block Accuracy
Ordered by Sequence No. in Methods File
Block format:

1 = Status
0-.5 = In control
>.5 = Warning
≥1 = Out of control
2,3,4 = coefficients of equation of upper and lower limit times
5 = number of points in file
6-63 = up to 58 data points.

STATSN:DA Statistics Data Base (one for each type of sample)
One block for each analysis
Block format:

1 = concentration of 1X x standard
2 = current detection limit
3 = number of data sets accumulated
4 = S^2 for precision
5,6,7 = coefficients for precision line
8 = S^2 for accuracy
9,10,11 = coefficients for accuracy line
12-25 = 2 level stack of results for each level, waiting for a complete set partial sums for each level
26-30 = 1/2X ΣX, ΣY, ΣXY, ΣX^2, ΣY^2

31-35 = 1X
36-40 = 2X
41-45 = 5X
46-50 = 10X
51-55 = 20X
56-60 = 50X
61 = $\Sigma Y-Y_I$
62 = $\Sigma(Y-Y_I)^2$

Parameter Files

METHOD:DA Methods File

Record 1
175 characters - Method Names (arranged according to RMA's analysis list
24 characters - 5 standard and SPEC sample names
10 characters - 5 standard sample type abbreviations

Record 2,3
100 numbers - Sequential analyses for analyses in each standard sample type
10 numbers - 2 x 5 array, start and size of string of analysis numbers in above array

Record 4
35 numbers - Analyses numbers for each lab.
6 numbers - 2 x 3 array, start and size of string of analysis numbers in above array

GRSIZE:DA Group Sample Size Table

Group Sizes

Loc	1	2	3	4	5
Value	21	16	11	6	1

Blanks

Loc	6	7	8	9	10
Value	3	2	2	1	1

Duplicates (or Replicates)

Loc	11	12	13	14	15
Value	4	3	3	2	2

Spikes

Loc	21	22	23	24	25
Value	7	6	5	4	3

Standards

Loc	16	17	18	19	20
Value	7	6	5	4	3

QCTIME:DA Q.C. Time File (one record for each sample type)

G(3) 1. Sample type

G(4) 2. Group Flag -1 = Non-grouped sample
0 = Initial state
N>0 = Grouped sample
N = Set in group

G(5) 3. Number of user samples remaining in group

G(6) 4. Pointer to position in sequence

G(7) 5. Next standard concentration

G(8) 6. Last location in present sequence
-63 sequence elements (up to 57)

OPERATION OF SYSTEM

This section outlines the procedures for use of the system, and is divided into sections for initial start up, normal operation, system status check, and G.C. operation. The sub-section illustrating system dialogue is given in Appendix C. The portion output by the system is underlined.

Initial Start-Up Procedure

This section describes the procedures to be used in starting up the system at first.

- Load all (or at least start up) programs and CSAVE under specified name.
- Using the CREATE command, create the following utility files:

STATUS:DA	40 records	(for 600 files)[a]
CREFT:DA	40 records	
METHOD:DA	4 records	
QCTIME:DA	5 records	
GRSIZE:DA	1 record	
TTAB90:DA	1 record	
TTABLE	3 records	
TEMFIL:DA	6 records	
SPCTEM:DA	6 records[b]	
STATS1:DA	16 records	
STATS2:DA	27 records	
STATS3:DA	27 records	
STATS4:DA	9 records	
STATS5:DA	21 records	
CUSUM1:DA	32 records[c]	

[a] The size of STATUS:DA and CREFT:DA can be chosen any desired size. 40 records was chosen as a suggestion.

[b] Not implemented on initial system.

[c] Only STATS3 and 4 and CUSUM 3 and 4 are needed at first, as the method of handling biological, soil, and sediment samples is undecided.

CUSUM2:DA	54 records
CUSUM3:DA	54 records
CUSUM4:DA	18 records
CUSUM5:DA	42 records

- Run set-up programs.

  WMTHOD:SV to write out METHOD:DA
  WTMFIL:SV to write out TEMFIL:DA
  WACTIM:SV to write out QCTIME:DA
  GRPSIZ:SV to write out GRSIZE:DA
  CRFLST:SV to initialize CREFT:DA
  FILCHK:SV to list sample files
  WSTATS:SV to initialize STATUS:DA
  WTABLE:SV to write out TTABLE:DA
  WTAB90:SV to write out TTAB90:DA

- Run standards on all analytes (at 1/2, 1, 2, 5, 10, 20, and 50 times detection limit[(a)]. (Ideally 4 replicates on each of 4 days.) Enter data into the system using USTATS:SV. (Can't use ENTERD yet, because it requires a cusum file. At least 4 replicates must be run before the next step can be taken.

- Run FXCSUM:SV. This takes the data from the statistical data base and creates both precision and accuracy cusum files.

Once FXCSUM:SV has been run for a given analysis type, the system is ready for normal operation

---

(a) A preliminary set of data should be run to determine the approximate value of the detection limit, unless it is already known.

Normal System Operation

Once the system has been started by the above procedure, normal operation involves a different sequence described below.

Sample Login

Samples are entered into the system by use of ENTERS:SV. The program assigns sequential laboratory numbers, indicating where blanks, duplicates, and standards occur. It also choses values for the standards. When all samples have been labled with the designated laboratory number and all blanks, duplicates, and standards are ready for analysis, and the indicated files have been created, the program FILWRT:SV is run. This program takes the information stored by ENTERS:SV and writes the analysis information into the files. It also enters the sample numbers into STATUS:DA and CREFT:DA. When samples are ready for analysis, the program ANALYZ:SV is run. This program checks the cusum files for the analysis. If the analysis is not in control, the program prints Q.C. OUT-OF-CONTROL, and will not accept samples for that analyte. If the analysis is in control, it sets the status flag for each analysis in each sample to "STARTED".

Laboratory Data Input

Data are entered into the sample files by use of the program ENTERD:SV. For user samples, the appropriate CUSUM file is queried to ascertain that the analysis is still in control, and if so, the data is enetered into the file, and the status flag for that analysis in that specific sample is set to done. If the analysis has gone out of control, the data is rejected.

For Q.C. samples, the appropriate CUSUM file is updated and a flag set to indicate status of the system. For standards or spiked data, the statistical data base is updated, and when one complete set of standards has been run, the precision, accuracy, and detection limit are re-evaluated. If the system is approaching the out-of-control point, a Q.C.

WARNING message is given. If it is out of control, a message, "Q.C. OUT-OF-CONTROL" is given and no further analyses can be started or data entered into sample files.

Q.C. Operations

When an analysis is flagged as out of control, or approaching the out-of-control point, the system will type out a warning message. At this time, it is necessary to assess the situation. The program CKCSUM:SV plots the CUSUM file data on one of the system terminals for on-the-spot diagnosis. If the analysis is out of control, a new statistical data base can be set up by running a new set of standards as was done to start the system.

DISCUSSION AND FUTURE WORK

The data management system described in the previous sections (and the Appendices) has been debugged as well as is practical in without real data. The system has been installed at RMA, and is currently in the process of acquiring data in the start-up phase, and should be operational in the next few weeks.

Actual experience with the system is essential in assessing the results and determining what problems exist. It is a virtual certainty that some program bugs will be uncovered in the early weeks of operation. However, these are areas which need further development before the system can be considered totally operational. In addition, changes in requirements for the system have been made that have not been put into effect yet because they occurred so late that incorporation in the initial system would have delayed it unduly. The known requirements are discussed in the following paragraphs.

Standards vs. Spikes

Working spiked samples into the computerized data management system is more difficult than for standard (known) samples. For water samples, standard samples are workable. However, no "standard" soil sediment of biological samples are available so that only real spiked samples seem appropriate. Accordingly, it was decided that the initial system would be concerned only with water samples. Tentative provisions have been made for spiked samples, but the exact means of implementation has not yet been decided upon.

Alteration of Data Format

When the system was being designed, the header block for the data file and the data block for each analyte were defined to be compatible with Edgewood's data file format. Therefore, analyte names were allotted 5 characters, the mantisso for the analysis was allotted 6 characters and the units file was allotted 3 characters. Similarly, the file header was provided with 2 characters for the data called "FILE", 2 characters (BA, SO, WA, SE, for Biological Assay, SOil, WAter, or SEdiment, respectively). In a recent visit to RMA it was learned that these requirements have been altered: 6 characters for analyte name, 5 characters for analysis mantisso, and 4 characters for units. In addition, the header needs an additional field "LOCATION (or WELL) IDENTIFIER" and doesn't need the 2 character "FILE". These changes have yet to be worked into the system.

Additional Programs

All of the originally planned programs have been implemented on the system with the exception of two that are intimately linked with the data format. These are the program to transmit data to the Tektronic 4051 for conversion to the proper format for the Edgewood file, and the program to list all samples ready for analysis in individual laboratories. These

remain to be completed. In addition, the Methods File will need to be modified as new sample types are defined, even for RMA. Installing the system at other labs will probably mean an entirely different Methods File. Accordingly, an EDITOR to modify the Methods File should be developed.

The conversion to the new data format will require about 1 man-week, and probably 1 man-week for debugging all the changes required. The new programs probably will require about 2 to 2-1/2 man-months for development and debugging. The time required for debugging the installed system is difficult to estimate, but could involve 1 to 2 man-months.

APPENDIX A

USE OF PROGRAMS

USTATS:SV Program to Build Statistics Data Base for Analytes

This program starts with an empty data base file, or one which has become suspect or worthless (e.g., when an analysis goes out of control). It requires only that a file named STATSn:DA exist on the system, (where n is the number of the sample types currently defined).

Sample Type

No.	Name	STATS in File Length
1	CPB (comprehensive pilot, biological)	16 records
2	CPS (comprehensive pilot, soil)	27 records
3	CPW (comprehensive pilot, water)	27 records
4	360 (360° water)	9 records
5	BSNF (Basin F)	21 records

and the methods file, METHOD:DA, which contains identification data for the sample and analysis type. When the program is started, it first checks for the availability of the methods file, and if it is not present, it aborts with error message (1).

(1) METHOD:DA NOT AVAILABLE

If the file is present, the program prints (2)

(2) ENTER SAMPLE TYPE

In response, the sample type is entered.
If it is not defined in the methods file, error message (3) is printed,

(3) ILLEGAL SAMPLE TYPE

and the program waits for a valid type. The program can be terminated by hitting the space bar and a carriage return. When an acceptable sample type is received, the program then looks for the proper STATSn:DA file. If not found, it issues the error message (4),

(4) STATSn:DA NOT AVAILABLE

and goes back to the request for a sample type, (2). If the stats file is found, the program then prints (5)

(5) ENTER ANALYSIS NAME

If the analysis is not included in the methods file, an error message (6) is printed,

(6) UNDEFINED ANALYSIS FOR THIS SAMPLE TYPE

and the prompt (5) is repeated. If the analysis is legal, the program then prints (7).

(7) NEW DATA BASE?

If the new data base is being built (whether for a totally new file or rebuilding a file that has gone bad, a YES (or Y) response will cause the system to write -1 and 0 on appropriate portions of the file, and to issue the prompt (8).

(8) ENTER THE 1X CONCENTRATION LEVEL

The response to this question is entered into the file and thereafter, all standards are expected to be related to this concentration.
If the response to the question (7) is NO, the program types out the 1X concentration level, and the status of the system, number of sets of data in the data base and contents of this temporary register. The system then types (9).

(9) ENTER TARGET LEVELS

The response must be 1/2, 1, 2, 5, 10, 20, or 50. Any other response will cause an error message (10) to be output.

(10) ILLEGAL TARGET LEVEL
ENTER TARGET LEVEL

Only two values can be present for each level. If temporary storage is already full, the error message (11) will be output and the system will go on to 0.

(11) LEVEL ALREADY FULL

After a satisfactory target level has been entered, the system will print (12).

(12) ENTER ANALYSIS RESULT

Any legal number will be accepted, so be careful.
If an erroneous number is entered and a carriage return is given before the error is discovered, it can be corrected at any later time by entering the complement (negative value) of the number in a later step. If a negative number is entered, the system will respond (12a).

(12a) ENTER REPLACEMENT VALUE?

If the erroneous number is still in the temporary storage register, it will simply be replaced and the system will continue at (15). If the value is not in the temporary storage register, the system will respond (12b).

(12b) VALUE NOT IN TEMPORARY STORAGE
DO YOU WANT DATA BASE CORRECTION?

A YES response will cause the system to correct the partial sums in the data base. A NO response will cause the system to ignore the negative input, and in either case, to procede on to (15).

If the temporary storage levels are full, the system will type (13).

(13) LEVEL NOW FULL

After each data point is entered, the system checks to see if one complete set of data has been entered into temporary storage. If so, these data are written to the permanent data base, and the temporary storage registers are again made available for use.

In any event, the system will then ask (14).

(14) DO YOU WANT STATISTICAL CALCULATIONS?

If the response is YES, the system will calculate and print out the detection limit and the precision and accuracy as a function of concentration. The least squares line for the latter two are also calculated. The detection limit, the variance and line coefficients for precision and accuracy, the updated partial sums are also rewritten in the statistical file (STATSn:DA) for later access by the program. Typical dialogue sequences are shown below.

(15) MORE DATA?

If the response is NO, the updated data are written into the system file and the program stops. If the answer is YES, the system asks (16).

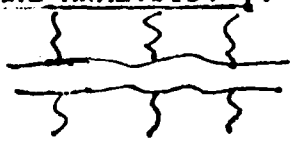
(16) SAME ANALYSIS?

If the response is YES, the program goes back to (9). If the response is NO, it updates the present data base, then starts over at step (2).

The following examples of output dialogue illustrate the operation of USTATS:SV. (System dialogue is underlined.)

The first example shows the creation of a new data base. Some steps have been omitted for brevity, but the process was carried far enough to permit calculation of statistics. (This file was created from replicate standards with identified analysis results, so the errors should all be zero.)

```
>RUN
ENTER SAMPLE TYPE: ?360
ENTER ANALYSIS NAME:?F
NEW DATA BASE??Y
ENTER IX CONCENTRATION LEVEL:?.5
ENTER TARGET LEVEL: ?.5
ENTER FOUND VALUE: ?.25
MORE DATA TO ENTER? ?Y
SAME ANALYSIS? ?Y
ENTER TARGET LEVEL: ?1
ENTER FOUND VALUE: ?.5
MORE DATA TO ENTER? ?Y
SAME ANALYSIS? ?Y
ENTER TARGET LEVEL: ?2
ENTER FOUND VALUE: ?1
MORE DATA TO ENTER? ?Y
SAME ANALYSIS? ?Y
ENTER TARGET LEVEL: ?5
ENTER FOUND VALUE: ?2.5
MORE DATA TO ENTER? ?Y
SAME ANALYSIS? ?Y
```



```
ENTER TARGET LEVEL: ?20
ENTER FOUND VALUE: ?10
MORE DATA TO ENTER? ?Y
SAME ANALYSIS? ?Y
ENTER TARGET LEVEL: ?50
ENTER FOUND VALUE: ?25
DO YOU WANT STATISTICS CALCULATED? ?Y
THE VALUE OF T IS 1.701
THE SLOPE IS 1
THE INTERCEPT IS 0
THE STANDARD DEVIATION IS 0
THE OVERALL COEFFICIENT OF VARIATION IS 0 %
 THE DECISION LIMIT IS 0
 THE DETECTION LIMIT IS -1
LEVEL          % VARIATION
.25              0
.5               0
1                0
2.5              0
5                0
10               0
25               0
LEVEL          % ACCURACY
.25              0
.5               0
1                0
2.5              0
5                0
10               0
25               0
MORE DATA TO ENTER??N

BASIC READY
```

The second example shows addition of data to an existing data base. In this case, the analytical results were allowed to vary so that results should not be zero.

```
>RUN
ENTER SAMPLE TYPE: ?360
ENTER ANALYSIS NAME:?CL
NEW DATA BASE??N
3 SETS OF DATA ARE IN THE DATA BASE.
THE 1X TARGET CONCENTRATION IS 1
TEMPORARY STORAGE STATUS:
LEVEL    CONTENTS
.5         .5
1          1
2          2
5          5
10         10
20         20
50
ENTER TARGET LEVEL: ?50
ENTER FOUND VALUE: ?50
DO YOU WANT STATISTICS CALCULATED? ?Y
THE VALUE OF T IS 1.701
THE SLOPE IS 1.07403
THE INTERCEPT IS 3.65998E-02
THE STANDARD DEVIATION IS 1.79103
THE OVERALL COEFFICIENT OF VARIATION IS 13.1545 %
 THE DECISION LIMIT IS 3.1682
 THE DETECTION LIMIT IS 5.8643
LEVEL       % VARIATION
.5            8.22596
1             8.22596
2             11.0867
5             9.57427
10            9.57427
20            9.57427
50            9.57427
LEVEL       % ACCURACY
.5            10.5
1             10.5
2             13.75
5             7.5
10            7.5
20            7.5
50            7.5
MORE DATA TO ENTER??N


BASIC READY
```

The third sample also shows addition to an existing data base, and also illustrates the use of a negative value to correct an erroneous entry.

```
ENTER SAMPLE TYPE: ?360
ENTER ANALYSIS NAME:?CL
NEW DATA BASE??N
3 SETS OF DATA ARE IN THE DATA BASE.
THE 1X TARGET CONCENTRATION IS 1
TEMPORARY STORAGE STATUS:
LEVEL    CONTENTS
.5         .5
1          1
2          2
5
10
20
50
ENTER TARGET LEVEL: ?5
ENTER FOUND VALUE: ?45
MORE DATA TO ENTER? ?Y
SAME ANALYSIS? ?Y
ENTER TARGET LEVEL: ?10
ENTER FOUND VALUE: ?10
MORE DATA TO ENTER? ?Y
SAME ANALYSIS? ?Y
ENTER TARGET LEVEL: ?5
ENTER FOUND VALUE: ?-45
ENTER REPLACEMENT VALUE?5
MORE DATA TO ENTER? ?Y
SAME ANALYSIS? ?Y
ENTER TARGET LEVEL: ?20
ENTER FOUND VALUE: ?20
MORE DATA TO ENTER? ?N
```

FXCSUM:SV Program to Setup Cumulative Sum File

This program takes data from the statistical data base created by USTATS:SV and builds the cumulative sum files for both precision and accuracy. It requires that a file, CUSUMn:DA, exist on the system, where n is the sample type. These files require no initialization. The STATSn:DA file, created by USTATS:SV, and the Methods File, METHOD:DA, are also required.

n	Sample Type	CUSUMn:DA Length
1	CPB	32
2	CPS	54
3	CPW	54
4	360	18
5	B	42

The program is called by the command: *RUN FXCSUM:SV*.
The system outputs (1)

(1) ENTER SAMPLE TYPE?

If the sample type is not correctly entered, the system responds with (2),

(2) NON EXISTENT SAMPLE TYPE

and repeats (1).
If the sample type is alright, the system checks for the CUSUMn:DA and STATSn:DA files. If either file is not found, the system prints (3),

(3) _____n:DA NOT AVAILABLE

and terminates. If both files are found, the system prints (4).

(4) ENTER ANALYTE NAME?

If the analyte name is incorrect, the system responds with (5),

(5) ANALYTE NOT IN THIS SAMPLE TYPE!

and repeats (4). If the analyte is correct, the system then procedes to write out the two records in CUSUMn:DA and terminates with (6)

(6) BASIC READY.

This sequence is illustrated in the following example dialogue.

In this example, the sample type BSNF did not have the data base completed for any analyte than Cl. Hence, Cl worked OK. F was rejected as it is not an analyte in BSNF samples. The END-OF-FILE message occurred because the data base record for NIT had not been prepared.

In the second example, the cusum file for CPW samples had not been created.

```
BASIC READY
>RUN FXCSUM
ENTER SAMPLE TYPE:?BSNF
ENTER ANALYTE NAME:?CL
ENTER ANALYTE NAME:?F
ANALYTE NOT IN THIS SAMPLE TYPE!
ENTER ANALYTE NAME:?NIT
END-OF-FILE/END-OF-RECORD IN LINE  620
>RUN
ENTER SAMPLE TYPE:?CPW
CUSUM3:DA NOT AVAILABLE!

BASIC READY
>
```

CKCSUM:SV - PROGRAM TO PLOT CUSUM FILE ON SYSTEM CONSOLE

This program plots the cusum file for a specified analyte in a specified sample type on the system console (or other specified device). Use of the program, of course, requires the presence of the appropriate CUSUMn:DA file (n is the sample type as defined on A-1), and the methods file METHOD:DA. It first checks for the availability of the methods file, and if it is not present, it aborts with the error message (1).

(1) METHOD:DA NOT AVAILABLE

If the file is present, the program prints (2)

(2) ENTER SAMPLE TYPE

and entering any character (other than space) followed by return will end the program.

If it is not defined in the methods file, error message (3) is printed,

(3) ILLEGAL SAMPLE TYPE

and the program waits for a valid type. The program can be terminated by hitting the space bar and a carriage return. When an acceptable sample type is recieved, the program then looks for the proper STATSn:DA file. If not found, it issues the error message (4),

(4) STATSn:DA NOT AVAILABLE

and goes back to the request for a sample type, (2). If the stats file is found, the program then prints (5)

(5) ENTER ANALYSIS NAME

If the analysis is not included in the methods file, an error message (6) is printed,

(6) UNDEFINED ANALYSIS FOR THIS SAMPLE TYPE

and the prompt (5) is repeated. Entering any character (other than space) followed by return will restart at (2).

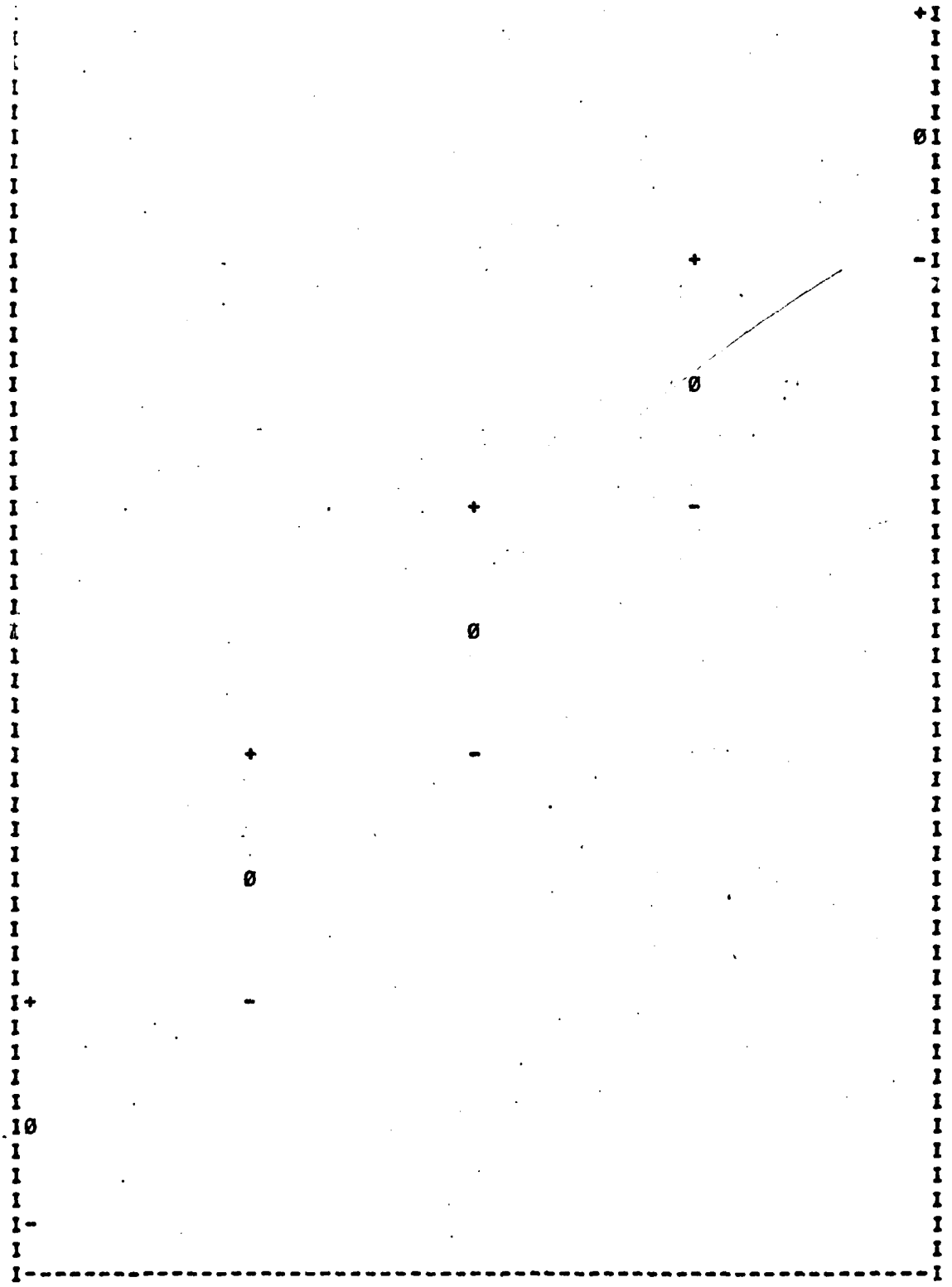
When a proper analyte is entered, the system prints (7)

(7) WHICH PART?

Acceptable responses are P, A, or E; P for precision cusum, A for accuracy cusum, and E for return for another analyte. In the plot on a terminal or line printer, the upper limit line is shown by +, the lower limit line by -, the expected line by 0, and the actual line by *. A typical example follows. (This plot was made from a new cusum chart containing no data, only the limits.

```
>RUN
ENTER SAMPLE TYPE? BSNF
ENTER ANALYSIS NAME? CL
WHICH PART? P
```

PRECISION CUSUM FILE FOR CL IN BSNF
JULY 9,1976



ENTERS:SV - PROGRAM TO ENTER SAMPLES INTO SYSTEM

This program is used to log samples into the data management system. Enters requires the presence of four data files:

- METHOD:DA - file of methods information
- GRSIZE:DA - file of group size data
- QCTIME:DA - file to store information about the sample group in process
- TEMFIL:DA - temporary file to store information about each sample for later permanent storage.

Enters recognizes the five sample types currently contained in the methods file. A definable sample class "SPEC" is planned but not yet implemented. It permits samples to be handled as GROUPED or UNGROUPED samples, depending upon the situation or viewed by the person responsible for entering samples.

GROUPED samples will normally be used when a relatively small number of samples is being handled. The intent is that a complete set of Q.C. samples will be included with each group. Some alteration in what constitutes a "complete set of Q.C. samples" may be required as experience is gained with the system, but the initial setup includes an entire set of 7 standards (0.5X to 50X) and duplicates or blanks. The group size table includes specification of Q.C. samples for any size group.

The UNGROUPED category is designed for a sample which will be run very often (50 or more per week). This category will include a set of 7 standards, 3 duplicates, and 3 blanks for each 30 samples.

Normally a GROUPED sample set may be started any time one GROUPED set has been completed, or if the system is running in the UNGROUPED mode. It is not necessary to enter all samples of a group at one time, although that can be done if desired. The system keeps track of each group in the file QCTIME:DA.

The only limitation is that TEMFIL:DA can only hold information for 60 samples, including the Q.C. samples.

The system is protected as much as possible. Enter of numeric data where alphabetic data is required (or vice versa) will cause the program to print ?? and wait for a proper response. There is one point of caution, however. The use of Control-Q MUST be avoided for termination, because the program requires proper exit, which involves writing data in three files. Use of the Control-Q will completely ruin sample number continuity and Q.C. spacing records.

When ENTERS is run, the program initially prompts (1).

(1) GROUPED SAMPLE?

If the response is N (or NO), the program prompts (2).

(2) ENTER SAMPLE TYPE?

If the sample type is not defined in METHOD:DA, the program prompts "NON-EXIST SAMPLE TYPE" and repeats prompt (2). When a suitable sample type is entered, the program checks QCTIME:DA. If an uncompleted group is currently in progress, the program prompts (3).

(3) GROUP CURRENTLY IN PROGRESS.
n REMAINING. SHOULD GROUP BE COMPLETED?

If the response is Y (or YES), the old group is completed. If the response is N (or NO), the current group is abandoned and this sample type is run in UNGROUPED mode until a new group is defined. In this case, the program then proceeds with actual sample entry dialogue (6).

If the response to (1) is Y (or YES), the program then prompts (4).

(4) NEW GROUP?

If the response is (Y), the program prompts (5).

(5) GROUP SIZE?

If the response is greater than 25, the program prompts "MAX GROUP SIZE IS 25" and repeats prompt (5). When a satisfactory group size has been enetered, the program then prompts (2).

If the sample type is acceptable and there is no current group, the program procedes to setup a sample group, then prompt (6).

(6) LAB. NO IS A00001 (or whatever the next number actually is)
n RECORDS NEEDED.
ENTER USER SAMPLE NO?

A six-character sample number provided by the user is entered.

The program then prompts (7)

(7) ENTER JULIAN DATE RECEIVED?

The 5-digit Julian date is entered. For each sample group at a single sample login session, the letter S can be used after a date for the first sample enetered. This will cause the same date to be repeated.

The program automatically assigns Q.C. samples in a pseudo-random order* and prints out the lab no. and the identifying data. Standards are given in terms of the concentration used in initially setting up the statistical data base. This concentration should not be changed (even if the detection limit does drift) until a new statistics data base is started, otherwise the validity of the accuracy data will be destroyed.

If no Q.C. samples are in order (or after the Q.C. sample information has been typed, the program then prompts (8).

(8) ANOTHER SAMPLE?

If the response is Y (or YES), the program proceeds to the next sample, prompt (6). If the response is N (or NO), the program then prompts (9).

(9) ANOTHER GROUP.

If the response is Y (or YES), the program returns to prompt (1).

If the response to (9) is N (or NO), the program terminates.

As long as room remains in TEMFIL:DA, ENTERS can be rerun repeatedly; the new entries are added at the end. When a group (including the Q.C. samples) is larger than the remaining space in TEMFIL:DA, a warning prompt is given: "THERE IS ROOM FOR ONLY N FILES". Also, when TEMFIL:DA is filled, the program will automatically end.

---

*A random number sequence is used, but it is required that duplicates must be duplicates of a user sample, not a blank or standard, so duplicates are not placed in a totally random manner.

Illustrating and largely self-explanatory dialogue is given below:

```
>RUN ENTERS
GROUPED SAMPLE? ?Y
NEW GROUP? ?N
ENTER SAMPLE TYPE: ?360

LAB. NO. IS A00009
2 RECORDS NEEDED.
ENTER USER SAMPLE NO:?A12678
ENTER JULIAN DATE RECEIVED. ?12378
ANOTHER SAMPLE?Y

LAB. NO. IS A00010
2 RECORDS NEEDED.
ENTER USER SAMPLE NO:?A12679
ENTER JULIAN DATE RECEIVED. ?S
ANOTHER SAMPLE?Y

LAB. NO. IS A00011
2 RECORDS NEEDED.
ENTER USER SAMPLE NO:?A12680
ENTER JULIAN DATE RECEIVED. ?S
ANOTHER SAMPLE?Y

LAB. NO. IS A00012
2 RECORDS NEEDED.
ENTER USER SAMPLE NO:?A123677
ENTER JULIAN DATE RECEIVED. ?S

LAB. NO. IS A00013
2 RECORDS NEEDED.
Q.C. SAMPLE: STANDARD- CONC.=.5X
ANOTHER SAMPLE?N
ANOTHER GROUP?Y
GROUPED SAMPLE? ?Y
NEW GROUP? ?N
ENTER SAMPLE TYPE: ?360
```

```
LAB. NO. IS A00014
2 RECORDS NEEDED.
ENTER USER SAMPLE NO:?BC1245
ENTER JULIAN DATE RECEIVED. ?03978
ANOTHER SAMPLE?N
ANOTHER GROUP?Y
GROUPED SAMPLE? ?Y
NEW GROUP? ?Y
GROUP SIZE? ?BSNF
??10
ENTER SAMPLE TYPE: ?BSNF
GROUP ALREADY IN PROGRESS!
5       REMAINING. SHOULD IT BE COMPLETED?Y

LAB. NO. IS A00015
4 RECORDS NEEDED.
ENTER USER SAMPLE NO:?QW1111
ENTER JULIAN DATE RECEIVED. ?04478

LAB. NO. IS A00016
3 RECORDS NEEDED.
Q.C. SAMPLE; DUPLICATE OF LAST SAMPLE!

LAB. NO. IS A00017
3 RECORDS NEEDED.
Q.C. SAMPLE: STANDARD- CONC.=5X
ANOTHER SAMPLE?N
ANOTHER GROUP?N

BASIC READY
```

FILWRT:SV - PROGRAM TO WRITE OUT SAMPLE FILES FROM TEMFIL:DA

FILWRT is used to write out sample files indicated when ENTERS is run. FILWRT accesses the methods file, METHOD:DA, writes the required sample information in the sample files, chains to DIRFIX:SV to enter the files in the status file, STATUS:DA, and the cross reference file, CREFT:DA. For proper operation, it is essential that:

(1) All sample files have been created, with the proper length.

(2) The statistics file for all analyses have been set up.

(3) There is sufficient room in STATUS:DA and CREFT:DA for all files to be entered.

If any of these conditions are not fulfilled, FILWRT will terminate abnormally with a diagnostic message.

FILWRT involves very little dialogue. When it starts, it types the prompt (1).

(1) ENTER NO. OF RECORDS TO SKIP?

This question should be answered with 0, unless one of the error conditions has occurred (discussed in more detail in subsequent paragraphs). It will then proceed to process TEMFIL:DA typing the name of each file in sequence. If it finishes normally, it types the message (2).

(2) ALL FILES IN TEMFIL:DA COMPLETED

If all files are written out, but there is not room in STATUS:DA or CREFT:DA, message (2) will be followed by messages (3) and (4) (Error A).

(3) STATUS:DA OR CREFT:DA FULL!

(4) ONLY n ENTRIES MADE FROM TEMFIL:DA

Two other error conditions can occur. If one of the sample files is missing, the program will type (5) followed by (4) (Error B).

(5) DATA FILE Xnnnnn:DA NOT AVAILABLE

If one of the statistics files are absent, message (6) will be typed followed by (4) (Error C).

(6) STATISTICS FILE STATSn:DA NOT AVAILABLE

Recovery from an error involves correcting the error and rerunning FILWRT. Errors B and C are corrected by creating the missing data file or

completing the incomplete statistics file. Error A requires either deleting inactive (completed) names from STATUS:DA and CREFT:DA or making the STATUS:DA and CREFT:DA longer. The deletion of file names is planned as a part of the program to transmit data to the Tektronix 4051 (page 21)

When rerunning FILWRT after the error has been ocrrected, enter the number typed out in the error message in response to prompt (1).

Sample dialogue sequences are shown below.

```
BASIC READY
>RUN ENTERS
GROUPED SAMPLE? ?Y
NEW GROUP? ?Y
GROUP SIZE? ?5
ENTER SAMPLE TYPE: ?360

LAB. NO. IS A00001
2 RECORDS NEEDED.
Q.C. SAMPLE-BLANK!
ANOTHER SAMPLE?Y

LAB. NO. IS A00002
2 RECORDS NEEDED.
ENTER USER SAMPLE NO:?A12345
ENTER JULIAN DATE RECEIVED. ?04278
ANOTHER SAMPLE?Y

LAB. NO. IS A00003
2 RECORDS NEEDED.
ENTER USER SAMPLE NO:?A12346
ENTER JULIAN DATE RECEIVED. ?3

LAB. NO. IS A00004
2 RECORDS NEEDED.
Q.C. SAMPLE; DUPLICATE OF LAST SAMPLE!
ANOTHER SAMPLE?N
ANOTHER GROUP?N

BASIC READY
>CREATE A00001:DA,2
>CREATE A00002:DA,2
>CREATE A00003:DA,2
>CREATE A00004:DA,2

>RUN FILWRT
ENTER NO. OF ENTRIES TO SKIP??
FILE A00001:DA
FILE A00002:DA
FILE A00003:DA
FILE A00004:DA
4       ENTRIES COMPLETED FROM TEMFIL:DA.
```

```
>RUN FILCHK
ENTER SAMPLE NUMBER?A00031
                                                    WA00030
0     1     A0030
                        0       0       0
CL                      0       0       0
F                       0       0       0
NIT                     0       0       0
PH                      0       0       0
SO4                     0       0       0
HARD                    0       0       0
NA                      0       0       0
DIMP                    0       0       0
DCPD
END OF SAMPLE.

ENTER SAMPLE NUMBER?A00032
0     2     A0030.2350 A12345.4876WA00030
                                                    01
CL                                                  02
F                                                   03
NIT                                                 04
PH                                                  05
SO4                                                 06
HARD                                                07
NA                                                  08
DIMP                                                09
DCPD
END OF SAMPLE.


ENTER SAMPLE NUMBER?

BASIC READY
>
```

ANALYZ:SV - Program to Start Analyses

This program is used when an analysis is started. It sets the appropriate flags in the analysis data file and the status file, so that system queries will indicate the actual status. It requires the methods file, METHOD:DA, the status file, STATUS:DA, and the corresponding CUSUM file. Absence of any one of these will cause a run-time error.

When the program is started, it issues the prompt (1).

(1) ENTER SAMPLE TYPE?

If an unknown sample type is entered, the system prints (2), and repeats (1), waining for an acceptable sample type.

(2) ILLEGAL SAMPLE TYPE!

When an acceptable sample type is entered, the system prompts (3).

(3) ENTER ANALYSIS NAME?

Any name not defined in METHOD:DA causes the error message (4), and repeats prompt (3).

(4) UNDEFINED ANALYSIS FOR THIS SAMPLE TYPE.

When an acceptable analysis name is enetered, the system checks the CUSUM file. If either the precision or accuracy file shows "out of control", the error message (5) is typed and the prompt (3) is repeated.

(5) PRECISION* Q.C. OUT OF CONTROL.

If the CUSUM file is OK, the system then types prompt (6).

(6) ENTER SAMPLE NØ?

If no such sample number is found, the prompt is repeated. If the sample file is found, the sample type is checked against the type entered in (1) above. If the types do not agree, the error message (7) is typed and repeats prompt (6).

(7) SAMPLE Xnnnnn IS WRONG SAMPLE TYPE?

If the sample type is OK, the system then checks the status flag for the analysis. If the analysis has already been started (or completed), the system issues an error message (8a) or (8b) and repeats prompt (6).

---

*or "ACCURACY", etc.

(8a) ANALYSIS ALREADY STARTED

(8b) ANALYSIS COMPLETED

If the analysis has not been started, the status flag is set to started in the sample file and in the status file.

The dialogue is illustrated in the following example:

CREF:SV - Program to Determine Status of Samples

This program returns the status of samples. It can be used to search for a sample by user number or by laboratory number. It requires the status file, the cross reference file and the analysis data file.

When started, the system prompts (1).

(1) ENTER USER SAMPLE?

Almost any six-character name is accepted.* The system reads the cross reference file, and if that entry is not found, issues the error message (2) and repeats (1).

(2) USER NO. NOT IN CROSS REFERENCE FILE!

If the entry is found, the system types (3),

(3) LABORATORY NO. IS Xnnnnn

and also indicates whether the sample has been started or not. If it has been started, it indicates how many analyses have been started and how many have been completed. It then types the prompt (4).

(4) DO YOU WANT THE STATUS OF INDIVIDUAL ANALYSES?

If the users response is N (or NO), the system goes to (1). If the users response is Y (or YES), the status of each analysis is printed individually, and the system then goes back to (1).

---

*CAUTION - EXCEPTION - Any character (other than space) followed by return is used if you want to get the status of a sample by lab. no. and will cause prompt (1a).

(1a) ENTER LABORATORY NUMBER?

In this case, step 3 is skipped, and the system proceeds to process the given lab. no.

The following dialogue illustrates the use of CREF:SV.

```
>RUN CREF
ENTER USER NO.: ?A12345
LAB. NO. IS A33031
ANALYSIS TYPE 360  NOT YET STARTED.
ENTER USER NO.: ?A12348
LAB. NO. IS A33035
ANALYSIS TYPE 360  NOT YET STARTED.
ENTER USER NO.: ?A1 1200
USER NO. NOT IN CROSS REFERENCE FILE!
ENTER USER NO.: ?
ENTER LAB. NO.?A33011
LAB. NO. NOT IN STATUS FILE!
ENTER USER NO.: ?@
ENTER LAB. NO.?@
```

ENTERD:SV - PROGRAM TO PUT ANALYSIS DATA IN FILE

ENTERD is used to put analysis results into the sample data files. Since, in theory, the analyst does not know which samples are user samples and which are Q.C. samples, the program treats all alike.

When the program is started, it issues the prompt (1).

(1) ENTER LAB SAMPLE NO:?

If the sample number entered does not correspond to an existing sample data file, an error message (2) is issued and prompt (1) is issued.*

(2) FILE UNNNNN:DA NOT AVAILABLE!

When a proper sample number has been entered, the program checks the sample data file heading for internal consistency. If no problems are found, it then issues the following series of prompts (3-9).

(3) ENTER ANALYSIS NAME:

(4) ENTER ANALYSIS RESULTS:

(5) ENTER DATE:

(6) ENTER ANALYSIS METHOD:

(7) ENTER INSTRUMENT NO:

(8) ENTER ANALYSTS INITIALS:

(9) ENTER MEASUREMENT UNITS:

If the analysis name does not exist, for the sample type, or is not recognized, the program issues error message (10), and goes back to (3).

(10) ANALYSIS NOT IN FILE.

If the analysis name is satisfactory, the next action depends on the type of sample.

For user samples, the program next checks the cusum files if either accuracy or precision is unsatisfactory, an error message (11) is issued and the program goes to step (12).

(11) PRECISION (or ACCURACY) Q.C. OUT OF CONTROL.

---

*A response of Control Q or any character followed by return aborts the program.

If the cusum files are satisfactory, prompt (12) is issued.

(12) ANOTHER ANALYSIS FOR THIS SAMPLE?

If the response is YES (or Y), it returns to prompt (3). If the response to (12) is NO (or N), the data are entered in the sample file, the status file is updated, and the program returns to prompt (1).

APPENDIX B

PROGRAM LISTINGS

INITIAL START-UP PROGRAMS

```
310 T=00004 IS ON CR32707 USING 00012 BLKS K=0010

  1  10REM - PROGRAM USTATS:SV
  2  20REM - PROGRAM TO UPDATE THE STATISTICS FILE
  3  30DIM A[0],B[2],C[2,7],D[7],E[7],F[7],G[7],H[7],I[7],M[4]
  4  40DIM D$[9]
  5  50DIM B$[24],C$[5],E$[9],A$[175],I[2,5],J[100]
  6  60DATA .5,1,2,5,10,20,50
  7  70LET S1=0
  8  80FOR I=1TO 7
  9  90READ Z[I]
  0  100NEXT I
  1  110LET D$="STATS :DA"
  2  120FILES*
  3  130ASSIGN "METHOD:DA",1,E
  4  140IF E=0GOTO 170
  5  150PRINT "METHOD:DA NOT AVAILABLE!"
  6  160STOP
  7  170READ #1;A$,B$[1,20]
  8  180READ #1,2
  9  190FOR I=1TO 100
 :0  200READ #1;J[I]
 :1  210NEXT I
 :2  220FOR I=1TO 2
 :3  230FOR J=1TO 5
 :4  240READ #1;I[I,J]
 :5  250NEXT J
 :6  260NEXT I
 :7  270PRINT "ENTER SAMPLE TYPE: ";
 :8  280INPUT C$[1,4]
 :9  290IF C$="    "STOP
 30  300LET J=-5
 31  310FOR I=1TO 5
 32  320LET J=J+4
 33  330IF C$=B$[J,J+3]THEN 370
 34  340NEXT I
 35  350PRINT "ILLEGAL SAMPLE TYPE!"
 36  360GOTO 280
 37  370CALLCHRS(48+I,D$[6,6])
 38  380ASSIGN D$,1,E
 39  390IF E=0THEN 420
 40  400PRINT D$," NOT AVAILABLE!"
 41  410GOTO 270
 42  420PRINT "ENTER ANALYSIS NAME:";
 43  430INPUT C$[1,5]
 44  440LET L=I[1,I]
 45  450LET I2=I[2,I]-1
 46  460FOR J=0TO I2
 47  470LET K=(J[J+L]-1)*5+1
 48  480IF C$=A$[K,K+4]THEN 520
 49  490NEXT J
 50  500PRINT "UNDEFINED ANALYSIS FOR THIS SAMPLE TYPE!"
 51  510GOTO 420
 52  520LET M[4]=J+1
 53  530LET S1=0
 54  540PRINT "NEW DATA BASE?";
 55  550GOSUB 1010
 56  560IF A1=2GOTO 650
```

```
58  590FOR I=1TO 20
59  590PRINT #1;-1
60  600NEXT I
61  610FOR I=25TO 63
62  620PRINT #1;0
63  630NEXT I
64  640GOTO 660
65  650LET S1=1
66  660READ #1,M[4]
67  670READ #1;M[1],M[2],M[3]
68  680FOR I=1TO 8
69  690READ #1;A[I]
70  700NEXT I
71  710IF M[3]<0LET M[3]=0
72  720IF M[1]>0GOTO 750
73  730PRINT "ENTER 1X CONCENTRATION LEVEL:";
74  740INPUT M[1]
75  750FOR I1=1TO 2
76  760FOR J=1TO 7
77  770READ #1;C[I1,J]
78  780NEXT J
79  790NEXT I1
80  800IF S1>0GOSUB 1750
81  810FOR J=1TO 7
82  820READ #1;D[J],E[J],F[J],G[J],H[J]
83  830NEXT J
84  840READ #1;B[1],B[2]
85  850REM - INPUT DATA FROM TERMINAL
86  860PRINT "ENTER TARGET LEVEL: ";
87  870INPUT X1
88  880FOR I1=1TO 7
89  890IF X1=Z[I1]THEN 930
90  900NEXT I1
91  910PRINT "ILLEGAL TARGET LEVEL!"
92  920GOTO 860
93  930IF C[2,I1]<0THEN 960
94  940PRINT "LEVEL ALREADY FULL!"
95  950GOTO 1340
96  960PRINT "ENTER FOUND VALUE: ";
97  970INPUT Y1
98  975IF Y1<0GOTO 1670
99  980IF C[1,I1]>0THEN 1010
00  990LET C[1,I1]=Y1
01  1000GOTO 1030
02  1010LET C[2,I1]=Y1
03  1020PRINT "LEVEL NOW FULL."
04  1030FOR I1=1TO 7
05  1040IF C[1,I1]<0GOTO 1340
06  1050NEXT I1
07  1060IF M[3]<16THEN 1090
08  1070LET F1=.9375
09  1080GOTO 1100
10  1090LET F1=1
11  1100LET L=M[1]
12  1110LET B[1]=B[1]*F1
13  1120LET B[2]=B[2]*F1
14  1130FOR I1=1TO 7
15  1140LET Y1=C[1,I1]
16  1150LET C[1,I1]=L[2,I1]
```

```
18  1170LET X1=Z[I1]*L
19  1180LET D[I1]=D[I1]*F1+X1
20  1190LET E[I1]=E[I1]*F1+Y1
21  1200LET F[I1]=F[I1]*F1+X1*X1
22  1210LET G[I1]=G[I1]*F1+Y1*Y1
23  1220LET H[I1]=H[I1]*F1+X1*Y1
24  1230LET T1=Y1-X1
25  1240LET B[1]=B[1]+T1
26  1250LET B[2]=B[2]+T1*T1
27  1260NEXT I1
28  1270LET M[3]=M[3]+1
29  1280IF M[3]<4GOTO 1340
30  1290PRINT "DO YOU WANT STATISTICS CALCULATED? ";
31  1300GOSUB 1610
32  1310GOTO A1OF 1320,1340
33  1320CHAIN "MUBVOS:SV"
34  1330GOTO 1700
35  1340PRINT "MORE DATA TO ENTER? ";
36  1350GOSUB 1610
37  1360IF A1=2GOTO 1730
38  1370PRINT "SAME ANALYSIS? ";
39  1380GOSUB 1610
40  1390IF A1=1GOTO 660
41  1400GOSUB 1440
42  1410PRINT "SAME SAMPLE TYPE?";
43  1420GOSUB 1610
44  1430GOTO A1OF 420,270
45  1440PRINT #1,M[4]
46  1450PRINT #1;M[1],M[2],M[3]
47  1460FOR I=1TO 6
48  1470PRINT #1;A[I]
49  1480NEXT I
50  1490FOR I1=1TO 2
51  1500FOR J=1TO 7
52  1510PRINT #1;C[I1,J]
53  1520NEXT J
54  1530NEXT I1
55  1540FOR I=1TO 7
56  1550PRINT #1;D[I],E[I],F[I],G[I],H[I]
57  1560NEXT I
58  1570PRINT #1;B[1],B[2]
59  1580RETURN
60  1590GOSUB 1610
61  1600GOTO A1OF 420,270
62  1610INPUT C$
63  1620IF C$[1,1]="Y"THEN 1660
64  1630IF C$[1,1]="N"THEN 1680
65  1640PRINT "PLEASE ENTER Y OR N!";
66  1650GOTO 1610
67  1660LET A1=1
68  1670RETURN
69  1680LET A1=2
70  1690RETURN
71  1700PRINT "MORE DATA TO ENTER?";
72  1710GOSUB 1610
73  1720GOTO A1OF 1370,2670
74  1730GOSUB 1440
75  1740STOP
76  1750PRINT M[3];TAB(0);" SETS OF DATA ARE IN THE DATA BASE."
```

```
77  1760PRINT "THE 1X TARGET CONCENTRATION IS ";TAB(0);M[1]
78  1770PRINT "TEMPORARY STORAGE STATUS:"
79  1780PRINT "LEVEL   CONTENTS"
80  1790FOR J=1TO 7
81  1800PRINT Z[J];TAB(10);
82  1810FOR I1=1TO 2
83  1820IF C[I1,J]<0GOTO 1830
84  1825PRINT C[I1,J];TAB(18);
85  1830NEXT I1
86  1840PRINT
87  1850NEXT J
88  1860RETURN
89  1870LET Y2=I1
90  1880PRINT "ENTER REPLACEMENT VALUE";
91  1890INPUT Y3
92  1900IF C[1,Y2]#-Y1GOTO 1930
93  1910LET C[1,Y2]=Y3
94  1920GOTO 1340
95  1930IF C[2,Y2]#Y1GOTO 1960
96  1940LET C[2,Y2]=Y3
97  1950GOTO 1340
98  1960PRINT "DATA NOT IN TEMPORARY STORAGE."
99  1970PRINT "DO YOU WANT DATA BASE CORRECTED";
00  1980GOSUB 1610
01  1990IF A1=2GOTO 1340
02  2000LET X1=X1*M[1]
03  2010LET E[Y2]=E[Y2]+Y3+Y1
04  2020LET G[Y2]=G[Y2]+Y3↑2-Y1↑2
05  2030LET H[Y2]=H[Y2]+X1*(Y1+Y3)
06  2040LET B[1]=B[1]+Y3+Y1
07  2050LET B[2]=B[2]+(Y3+Y1)↑2
08  2060GOTO 1340
09  2070END
```

```
:** T=00000 IS ON LU 05

   10REM - HUBVUS:SV
   20REM - PROGRAM TO TAKE DATA FROM CORE AND
   30REM - CALCULATE DETECTION LIMIT, PRECISION
   40REM - AND ACCURACY DATA, THEN WRITE IT IN A FILE.
   50COM A[8],I[2],C[2,7],D[7],E[7],F[7],G[7],H[7],Z[7],M[4]
   60COM D$[9]
   70DIM X[31],B[7]
   80LET N=W=M[3]*7
   90FILES TTAB90:DA
  100LET I=T=0
  110FOR K=1TO 31
  120READ #1;X[K]
  130NEXT K
  140LET K=N
  150IF K>31LET K=31
  160LET T=X[K]
  170PRINT "THE VALUE OF T IS ";TAB(0);T
  180ASSIGN D$,1,E
  190LET A1=A2=C1=C2=C3=0
  200FOR I=1TO 7
  210LET A1=A1+D[I]
  220LET A2=A2+F[I]
  230LET C1=C1+E[I]
  240LET C2=C2+H[I]
  250LET C3=C3+G[I]
  260NEXT I
  270GOSUB 850
  290LET S1=SQR((Y1-Z1↑2/X1)/(W-2))
  300PRINT "THE SLOPE IS ";TAB(0);M1
  310PRINT "THE INTERCEPT IS ";TAB(0);B1
  320PRINT "THE STANDARD DEVIATION IS ";TAB(0);S1
  330LET S2=100*S1/(C1/W)
  340PRINT "THE OVERALL COEFFICIENT OF VARIATION IS ";
  350PRINT TAB(0);S2;TAB(0);" %"
  355GOSUB 730
  360PRINT "LEVEL       % VARIATION"
  370LET M2=M[1]
  380FOR I=1TO 7
  390LET L=M2*Z[I]
  400LET B[I]=100*(SQR((G[I]-E[I]↑2/M[3])/(M[3]-1))/L)
  410PRINT L;TAB(15);B[I]
  420NEXT I
  430GOSUB 910
  440LET A[1]=S1↑2
  450LET A[2]=B1
  460LET A[3]=M1
  470LET A[4]=0
  480PRINT "LEVEL       % ACCURACY"
  490FOR I=1TO 7
  500LET L=M2*Z[I]
  510LET L0=L*M[3]
  515LET B[I]=100*(E[I]-L0)/L0
  520PRINT L;TAB(15);B[I]
  530NEXT I
  540GOSUB 910
  550LET A[5]=(I[2]-I[1]↑2/W)/(W-1)
```

```
590PRINT #1,M[4];M[1],M[2],M[3]
600FOR I=1TO 8
610PRINT #1;A[I]
620NEXT I
630FOR I1=1TO 2
640FOR J=1TO 7
650PRINT #1;C[I1,J]
660NEXT J
670NEXT I1
680FOR J=1TO 7
690PRINT #1;D[J],E[J],F[J],G[J],H[J]
700NEXT J
710PRINT #1;I[1],I[2]
720CHAIN "DSTATS:SV",1000
730LET E1=A1/W
740LET K1=B1+S1*T*(SQR(1+1/N+(L1*E1)/X1))
750LET S=S1*T
760LET C0=M1/S
770LET D=(K1-B1)/S
780LET A=1-X1*C0↑2
790LET B=2*(C0*D*X1+L1)
800LET C=E1↑2+X1*(1+1/N-D↑2)
810LET M[2]=L1=(-B-SQR(B↑2-4*A*C))/(2*A)
820PRINT " THE DECISION LIMIT IS ";TAB(0);K1
830PRINT " THE DETECTION LIMIT IS ";TAB(0);L1
840RETURN
850LET X1=A2-A1*A1/W
860LET Y1=C3-C1↑2/W
870LET Z1=C2-A1*C1/W
880LET M1=Z1/X1
890LET B1=(C1*A2-A1*C2)/(W*X1)
900RETURN
910LET A1=A2=C1=C2=C3=0
920FOR I=1TO 7
930LET L=M2*Z[I]
940LET A1=A1+L
950LET A2=A2+L↑2
960LET B0=B[I]
970LET C1=C1+B0
980LET C2=C2+B0*L
990LET C3=C3+B0↑2
1000NEXT I
1010GOSUB 850
1020RETURN
1030END
```

INSERT 745 IF S1=0 GO TO 820

INSERT 815 GO TO 830
820 M(2)=L1=0

delete - meaningless

```
UG12 T=00004 IS ON CR3270 USING 00012 BLKS K=0016

01   10REM - PROGRAM EXCSUM:SV
02   20REM - PROGRAM TO SET UP CUSUM FILES
03   30REM - FILE NAMES ARE CUSUM1:DA THROUGH
04   40REM - CUSUM5:DA FOR THE FIVE SAMPLE TYPES.
05   50REM - THIS PROGRAM REQUIRES THAT THE STATSN:DA
06   60REM - FILE ARE AVAILABLE
07   70REM - VERSION DATE 2/4/70
08   80REM - CUSUMN:DA RECORD LENGTHS ARE, RESPECTIVELY
09   90REM - N=1,32;=2,54;=3,54;=4,18;=5,42
10   100DIM A$[9],B$[9],D$[175],E$[20],B[100],C[2,5]
11   110DIM C$[5]
12   120FILESMETHOD:DA
13   130READ #1;D$,E$[1,20]
14   140READ #1,2
15   150FOR I=1TO 100
16   160READ #1;B[I]
17   170NEXT I
18   180FOR I=1TO 2
19   190FOR J=1TO 5
20   200READ #1;C[I,J]
21   210NEXT J
22   220NEXT I
23   230REM - ALPHA, BETA AND SYSTEM VARIABILITY PARAMETERS.
24   240DATA .05,.05,.2
25   250READ A1,B1,D
26   260LET A$="CUSUM :DA"
27   270LET B$="STATS :DA"
28   280FILES*,*
29   290PRINT "ENTER SAMPLE TYPE:";
30   300INPUT C$[1,5]
31   310IF C$[2,5]="    "STOP
32   320LET A$[0,0]=C$
33   330LET K=-3
34   340FOR I=1TO 5
35   350LET K=K+4
36   360IF C$[1,4]=E$[K,K+3]GOTO 400
37   370NEXT I
38   380PRINT "NONEXISTENT SAMPLE TYPE!"
39   390GOTO 290
40   400CALLCHR$(I+48,A$[0,0])
41   410CALLCHR$(I+48,B$[0,0])
42   420ASSIGN A$,1,E
43   430IF E=0THEN 480
44   440PRINT A$;
45   450GOTO 490
46   460ASSIGN B$,2,E
47   470IF E=0THEN 510
48   480PRINT B$;
49   490PRINT " NOT AVAILABLE!"
50   500STOP
51   510PRINT "ENTER ANALYTE NAME:";
52   520INPUT C$[1,5]
53   530LET I1=0
54   540LET K1=C[1,I]
55   550LET K2=C[2,I]
56   560FOR K=K1TO K1-1+K2
57   570LET J=B[K]
```

SAMPLE LOGIN PROGRAMS

```
10REM - PROGRAM ENTERS:SV
20REM - PROGRAM TO ACCEPT SAMPLE INFORMATION,
30REM - ASSIGN LABORATORY NUMBER, ISSUE DIRECTIONS
40REM - FOR QC SAMPLES AND WRITE OUT FILES.
50REM - 2/5/78 VERSION.
60DIM G[11],B[64],F[4],P[4],C[7],L[10]
70DIM A[50],S$[4],B$[24]
80DATA .5,2,10,50,1,5,20
90DATA 2,3,5,4
100LET G[11]=0
110LET G[10]=0
120FOR I=1TO 7
130READ C[I]
140NEXT I
150FILESMETHOD:DA,GRSIZE:DA
160READ #1;B$[1,24],B$[1,24]
170READ #1,4
180FOR I=1TO 41
190READ #1;A
200NEXT I
210FOR I=1TO 10
220READ #1;L[I]
230NEXT I
240ASSIGN "QCTIME:DA",1,E
250IF E=0GOTO 280
260PRINT "QCTIME:DA NOT AVAILABLE!"
270STOP
280LET G[2]=-1
290PRINT "GROUPED SAMPLE? ";
300INPUT S$[1]
310IF S$[1]="N"THEN 440
320IF S$[1]#"Y"THEN 290
330PRINT "NEW GROUP? ";
340INPUT S$[1]
350IF S$[1]="N"THEN 420
360IF S$[1]#"Y"THEN 330
370PRINT "GROUP SIZE? ";
380INPUT G[1]
390IF G[1]<= 25THEN 430
400PRINT "MAX. GROUP SIZE IS 25!"
410GOTO 370
420LET G[2]=G[2]+1
430LET G[2]=G[2]+1
440PRINT "ENTER SAMPLE TYPE: ";
450INPUT S$[1,4]
460LET K=1
470FOR I=1TO 6
480IF B$[K,K+3]=S$THEN 530
490LET K=K+4
500NEXT I
510PRINT "NONEXISTENT SAMPLE TYPE!"
520GOTO 440
530LET G[3]=I
540IF 1=0CHAIN "ENTRSP:SV"
550READ #1,1;G[3],G[4]
560READ #1;G[5],G[6],G[7],G[8]
570FOR L=1TO G[8]
580READ #1;B[L]
```

```
590NEXT L
600IF G[3]#ITHEN 640
610IF G[2]>0GOTO 660
620IF G[4]>0GOTO 650
630GOTO 760
640PRINT "HEADER ERROR. SAMPLE TYPE AND HEADER DIFFER."
650STOP
660IF G[4]>0GOTO 1010
670PRINT "NO GROUP IN PROGRESS;"
680GOTO 280
690PRINT "GROUP ALREADY IN PROGRESS; "
700PRINT G[5];" REMAINING. SHOULD IT BE COMPLETED";
710INPUT S$[1]
720IF S$[1]="Y"THEN 1010
730IF S$[1]="N"THEN 760
740PRINT "??"
750GOTO 710
760IF G[2]=0GOTO 820
770LET G[1]=25
780IF G[6]>0GOTO 1010
790LET G[1]=25
800REM - NEW GROUP.
810IF ABS(G[2])=1LET G[4]=G[2]
820IF G[4]=0LET G[4]=1
830FOR I=1TO 5
840READ #2;E[I]
850NEXT I
860IF G[1]>= E[1]THEN 880
870GOTO 820
880READ D0,D0,S0
890REM - CHECK SAMPLE TYPE; N1=3 OR 4 IS A WATER SAMPLE.
900IF G[3]<3OR G[3]>4READ S0
910REM - CALCULATE TOTAL NUMBER OF SAMPLES.
920LET K1=E[00]
930LET K2=E[00]
940LET K3=E[50]
950LET X1=G[1]
960LET T0=X1+K1+K2+K3
970FOR K=1TO K1
980LET B[K]=1
990NEXT K
1000FOR I=KTO K+K3-1
1010LET B[I]=3
1020NEXT I
1030LET T1=I+X1-1
1040FOR K=ITO T1
1050LET B[K]=0
1060NEXT K
1070FOR I=KTO T0
1080LET B[I]=2
1090NEXT I
1100REM - USE RND TO SET UP RANDOM PATTERN.
1110LET I=RND(-TIM(-2))
1120FOR I=1TO T0
1130LET A[I]=RND(1)
1140NEXT I
1150LET G[5]=X1
1160LET G[6]=1
1170LET G[0]=T0
1180REM - ROUTINE TO SORT ARRAY A
```

```
1190REM - IN INCREASING NUMERICAL ORDER, AND
1200REM - CARRY ALONG THE IDENT ARRAY B.
1210REM - T1 IS THE ARRAY LENGTH.
1220FOR J=1TO T1-1
1230FOR I=J+1TO T1
1240IF A[I]>= A[J]THEN 1310
1250LET T=A[J]
1260LET A[J]=A[I]
1270LET A[I]=T
1280LET T=B[J]
1290LET B[J]=B[I]
1300LET B[I]=T
1310NEXT I
1320NEXT J
1330REM - MAKE CERTAIN THAT DUP FOLLOWS REAL SAMPLE.
1340FOR K1=T1+1TO T0
1350LET A[K1]=INT(A[K1]*X1+.5)
1360IF A[K1]=0LET A[K1]=1
1370LET J1=1
1380FOR K=1TO T1
1390IF B[K]#0GOTO 1520
1400IF A[K1]#J1GOTO 1510
1410IF B[K+1]=2GOTO 1520
1420LET T1=T1+1
1430LET K=K+1
1440FOR J2=T1TO KSTEP -1
1450LET B[J2]=B[J2-1]
1460NEXT J2
1470IF B[K-1]=0GOTO 1500
1480LET A[K1]=INT(A[K1]/2)
1490GOTO 1370
1500LET B[K]=2
1510LET J1=J1+1
1520NEXT K
1530NEXT K1
1540REM - INSERT PROPER VALUE FOR STANDARDS.
1550FOR I=1TO T0
1560IF B[I]#3GOTO 1600
1570LET B[I]=G[7]
1580LET G[7]=G[7]+1
1590IF G[7]=10LET G[7]=3
1600NEXT I
1610CHAIN "ENTRS2:SV"
1620END
```

```
10REM - PROGRAM ENTRS2:SV
20REM - INPUT USER SAMPLE NUMBERS, ASSIGN
30REM - LABORATORY NUMBER, AND ENTER IN
40REM - QUE FILE
50REM - 2/14/76 VERSION.
60DIM G[11],B[64],F[4],P[4],C[7],L[10]
70DIM Z$[22],X$[5]
80FILES TEMP:IL:DA,STATUS:DA
90LET M1=G[6]
100IF G[11]>0GOTO 120
110READ #2;F[1],F[2],F[3],F[4]
120LET Z1=10
130GOSUB 1050
140GOTO 150
150READ #1;P[1],P[2],P[3],P[4]
160READ #1,P[1]
170IF P[2]=0GOTO 220
180FOR I=1TO P[2]
190READ #1;Z$[1,22]
200NEXT I
210GOTO 230
220LET Z$[1,22]="                    "
230LET G[11]=1
240IF G[1]<= P[3]GOTO 410
250PRINT "THERE IS ROOM FOR ONLY ";TAB(0);P[3];" FILES.";
260PRINT " IN TEMPORARY FILE!"
270GOTO 410
280IF B[M1]=0GOTO 320
290PRINT "ANOTHER SAMPLE";
300GOSUB 1700
310IF A1>0GOTO 1270
320GOSUB 340
330GOTO 410
340LET F[2]=F[2]+1
350IF F[2]<100@@GOTO 400
360LET F[2]=F[2]-1
370LET F[1]=F[1]+1
380IF (F[1]-INT(F[1]/1@0)*100)<9RETURN
390LET F[1]=F[1]+69
400RETURN
410LET F5=INT(F[1]/100)
420CALL CHR$(F5,Z$[1])
430LET F5=F[1]-F5*100
440CALL CHR$(48+F5,Z$[2,2])
450LET F5=F[2]/1000
460FOR I=3TO 6
470LET F6=INT(F5+.05)
480CALL CHR$(48+F6,Z$[I,I])
490LET F5=(F5-F6)*10
500NEXT I
510IF G[11]>0GOTO 600
520IF B[M1]<2GOTO 600
530FOR I=M1+1TO G[6]
540IF B[I]<>0GOTO 590
550LET T1=B[M1]
560LET B[M1]=B[I]
570LET B[I]=T1
580GOTO 600
```

```
600PRINT
610PRINT "LAB. NO. IS ";Z$[1,6]
620LET G[10]=1
630LET L1=0
640IF B[M1]=0THEN 680
650IF B[M1]>1THEN 700
660LET L1=5
670GOSUB 850
680PRINT "W.C. SAMPLE-BLANK!"
690GOTO 770
700LET L1=5
710GOSUB 850
720IF B[M1]>2THEN 750
730PRINT "W.C. SAMPLE; DUPLICATE OF LAST SAMPLE!"
740GOTO 770
750PRINT "W.C. SAMPLE: STANDARD- CONC.=";
760PRINT TAB(0);G[B[M1]-2];TAB(0);"X"
770LET Z$[7,11]="     "
780CALLCHR$(B[M1]+48,Z$[12,12])
790GOTO 930
800GOSUB 850
810PRINT "ENTER USER SAMPLE NO:";
820INPUT Z$[7,12]
830LET G[5]=B[5]-1
840GOTO 870
850PRINT L[L1+G[5]];TAB(0);" RECORDS NEEDED."
860RETURN
870PRINT "ENTER JULIAN DATE RECEIVED. ";
880INPUT Z$[13,17]
890IF Z$[13,13]="S"GOTO 920
900LET X$=Z$[13,17]
910GOTO 930
920LET Z$[13,17]=X$
930LET M1=0
940CALLCHR$(B[5]+48,Z$[18,18])
950CALLCHR$(M1+48,Z$[19,19])
960IF B[M1]=0GOTO 1040
970IF B[M1]=2GOTO 1000
980IF B[M1-1]=0GOTO 1010
990LET M1=M1+1
1000GOTO 1010
1010LET R1=2
1020LET W2=2
1030GOTO 1090
1040IF M1+1>B[0]GOTO 1080
1050IF B[M1+1]=2GOTO 1060
1060LET R1=1
1070GOTO 1120
1080LET W2=R1=0
1090CALLCHR$(R1+48,Z$[20,20])
1100CALLCHR$(W2+48,Z$[21,21])
1110CALLCHR$(48,Z$[22,22])
1120PRINT M1;Z$[1,22]
1130LET P[5]=P[5]-1
1140LET P[2]=P[2]+1
1150IF P[2]<1THEN 1220
1160LET P[2]=0
1170LET P[1]=P[1]+1
1180IF P[1]<7GOTO 1210
```

```
1200GOTO 1000
1210PRINT #1,P[1]
1220LET M1=M1+1
1230IF M1<= G[0]GOTO 200
1240LET G[0]=G[4]=0
1250M1F G[2]<0GOTO 1340
1260PRINT "GROUP COMPLETED."
1270GOSUB 540
1280PRINT "ANOTHER GROUP";
1290GOSUB 1700
1300IF A1>0GOTO 1300
1310GOSUB 1390
1320GOSUB 1030
1330CHAIN "ENTERS:SV",110
1340GOSUB 1390
1350CHAIN "ENTERS:SV",000
1360GOSUB 1390
1370GOSUB 1500
1380STOP
1390ASSIGN "GCTIME:DA",1,E
1400IF E=0GOTO 1430
1410PRINT "GLTIME:DA NOT AVAILABLE!"
1420RETURN
1430PRINT #1,G[3];G[3],G[4],G[5],G[6],G[7],G[8]
1440IF G[8]=0RETURN
1450FOR I=1TO G[0]
1460PRINT #1;B[I]
1470NEXT I
1480RETURN
1490REM - NOW UPDATE CURRENT LAB NO IN STATUS FILE.
1500FOR I=3TO 64
1510READ #2;B[I]
1520NEXT I
1530PRINT #2,1
1540PRINT #2;F[1],F[2],F[3],F[4]
1550FOR I=5TO 64
1560PRINT #2;B[I]
1570NEXT I
1020REM - FINALLY SAVE TEMFIL POINTERS.
1630ASSIGN "TEMFIL:DA",1,E
1640IF P[1]=1GOTO 1660
1650LET Z1=10
1660GOSUB 1830
1670GOTO 1760
1680IF P[2]=0GOTO 1720
1690LET Z1=P[2]
1700GOSUB 1830
1710IF P[2]=1000TO 1760
1720LET Z1=P[2]+1
1730FOR I=Z1TO 10
1740PRINT #1;"                    "
1750NEXT I
1760PRINT #1;P[1],P[2],P[3],P[4]
1770RETURN
1780INPUT U$
1790LET A1=0
1800IF U$[1,1]="Y"RETURN
1810LET A1=1
1820IF U$[1,1]="N"RETURN
```

```
1640GOTO 1/80
1650READ #1,I
1660FOR I=1TO 21
1670READ #1?2$[1,22]
1680NEXT I
1690RETURN
1900END
```

```
 4  T=00004 IS ON CR32707 USING 00012 BLKS R=0016

   10REM - PROGRAM FILWRT:SV
   20REM - PROGRAM TO SET UP SAMPLE FILES.
   30REM - VERSION DATE 2/24/78
   40COM J[5]
   50COM A$[175],B$[24],C[100],D[2,5],C$[4],T$[10]
   60DIM D$[12],N$[9],M$[9],Z$[240],E[7],Y$[9],V$[20]
   70PRINT "ENTER NO. OF ENTRIES TO SKIP";
   80INPUT I0
   90LET J[2]=I0
  100LET J[1]=0
  110LET J=-9
  120FOR I=1TO 24
  130LET J=J+10
  140LET Z$[J,J+9]="          "
  150NEXT I
  160LET N$[1,9]="       :DA"
  170LET M$="STATS :DA"
  180FILESME(MOD:DA,*,*
  190READ #1;A$,B$
  200LET T$="BASURAWASE"
  210FOR I=1TO 100
  220READ #1;C[I]
  230NEXT I
  240FOR I=1TO 2
  250FOR J=1TO 5
  260READ #1;D[I,J]
  270NEXT J
  280NEXT I
  290DATA .5,2,10,50,1,5,20
  300FOR I=1TO 7
  310READ E[I]
  320NEXT I
  330ASSIGN "TEMPIL:DA",1,E
  340IF E=0THEN 360
  350GOTO 1300
  360FOR A1=1TO 6
  370IF END #1THEN 1550
  380READ #1,A1
  390FOR A2=1TO 10
  400IF END #1THEN 1550
  410READ #1;V$[1,22]
  420IF V$[1,6]="      "GOTO 1550
  430IF I0=0GOTO 460
  440LET I0=I0-1
  450GOTO 1310
  460LET R0=1
  470LET F0=0
  480LET Z$[1,0]=V$[1,0]
  490LET N$[1,0]=V$[1,0]
  500ASSIGN #3,2,E
  510PRINT "FILE ";N$
  520IF E#0GOTO 1440
  530LET C$=N$[1,1]
  540LET A1=NUM(C$)*100
  550LET C$=N$[2,2]
  560LET A0=A1+NUM(C$)-48
  570LET A1=0
  580FOR I=3TO 6
```

```
590LET A1=A1*10
600LET C$=M$(I,I)
610LET A1=ASC(C$)-48+A1
620NEXT I
630LET C$=V$(10,10)
640LET N1=ASC(C$)-48
650LET I0=(N1-1)*4+1
660LET I0=(N1-1)*2+1
670LET Z$(11,21)=V$(7,17)
680LET Z$(22,25)=T$(16,I6+1)
690LET I5=D(1,N1)
700LET Z$(24,26)=V$(19,21)
710LET Z$(7,16)=M$(I5,15+3)
720LET Z$(27,28)="00"
730LET I2=D(2,N1)
740IF Z$(11,15)="      "GOTO 1010
750LET D$=40
760LET J=25
770LET D2=241
780FOR I=1TO I2
790IF F0=0GOTO 810
800GOSUB 940
810LET I4=C(I3+1-1)
820LET I4=(I4-1)*5+1
830LET Z$(J+1,J+3)=A$(I4,I4+4)
840LET Z$(J+0,J+35)="                                   0"
850LET X1=INT(I/10)*10
860LET X=(I-X1)
870CALLCHR$(X1+48,Z$(J+39,J+39))
880CALLCHR$(X+48,Z$(J+40,J+40))
890GOSUB 1300
900NEXT I
910LET Z$(J+1,J+4)="$$$$"
920IF J>0GOSUB 940
930GOTO 1250
940IF K0>1GOTO 970
950PRINT #2,1;A$,A1,Z$(1,20),Z$(29,228)
960GOTO 980
970PRINT #2,K0;Z$
980LET K0=K0+1
990RETURN
1000REM - HERE FOR Q.C.
1010LET J=19
1020LET D$=10
1030LET D2=126
1040LET M$(5,6)=C$
1050MASSIGN #3,5,E
1060IF E=0GOTO 1400
1070PRINT #2;A$,A1,Z$(1,28)
1080LET S$=V$(12,12)
1090LET N1=ASC(S$)-50
1100LET Z$(5,16)="                0"
1110FOR I=1TO 12
1120IF F0=0GOTO 1160
1130LET K0=K0+1
1140PRINT #2,K0
1150LET F0=0
1160LET I4=C(I5+1-1)
1170LET I4=(I4-1)*5+1
1180LET Z$(1,5)=A$(I4,I4+4)
```

```
 59 LET A1=A1*10
 60 LET C$=A$(I,I)
 61 LET A1=NUM(C$)-48+A1
 62 NEXT I
 63 LET C$=V$(10,18)
 64 LET N1=NUM(C$)-48
 65 LET I0=(N1-1)*4+1
 66 LET I0=(N1-1)*2+1
 67 LET Z$(11,21)=V$(7,17)
 68 LET Z$(22,26)=T$(10,I0+1)
 69 LET I0=D(1,N1)
 70 LET Z$(24,26)=V$(19,21)
 71 LET Z$(7,10)=B$(I0,I0+3)
 72 LET Z$(27,28)="00"
 73 LET I2=D(2,N1)
 74 IF Z$(11,15)="      "GOTO 1010
 75 LET D0=40
 76 LET J=26
 77 LET D2=241
 78 FOR I=1TO I2
 79 IF F0=0GOTO 810
 80 GOSUB 940
 81 LET I4=L(I0+I-1)
 82 LET I4=(I4-1)*5+1
 83 LET Z$(J+1,J+5)=A$(I4,I4+4)
 84 LET Z$(J+0,J+36)="                                    0"
 85 LET X1=INT(I/10)*10
 86 LET X=(I-X1)
 87 CALLCHR$(X1+48,Z$(J+39,J+39))
 88 CALLCHR$(X+48,Z$(J+40,J+40))
 89 GOSUB 1300
 90 NEXT I
 91 LET Z$(J+1,J+4)="$$$$"
 92 IF J>0GOSUB 940
 93 GOTO 1250
 94 IF R0>1GOTO 970
 95 PRINT #2,1;A0,A1,Z$(1,20),Z$(29,228)
 96 GOTO 980
 97 PRINT #2,R0;Z$
 98 LET R0=R0+1
 99 RETURN
100 REM = HERE FOR Q.C.
101 LET J=19
102 LET D0=10
103 LET D2=120
104 LET A$(5,6)=C$
105 MAT ION A$,J,E
106 IF E=0GOTO 1400
107 PRINT #2,A$,A1,Z$(1,28)
108 LET S$=V$(12,12)
109 LET N1=NUM(S$)-50
110 LET Z$(5,16)="                0"
111 FOR I=1TO 12
112 IF F0=0GOTO 1100
113 LET R0=R0+1
114 PRINT #2,R0
115 LET F0=J
116 LET I4=L(I0+I-1)
117 LET I4=(I4-1)*5+1
```

```
19   1150GOSUB 1360
20   1200LET V1=0
21   1210IF W1<= 1GOTO 1270
22   1220LET E=TYP(-5)
23   1230IF E=36OTO 1460
24   1240IF E=46OTO 1460
25   1250READ #3,I;D1
26   1260LET V1=E[W1]*D1
27   1270PRINT #2;Z$[1,16],V1,0,I
28   1280NEXT I
29   1290PRINT #2;END
30   1300LET J[1]=J[1]+1
31   1310NEXT K2
32   1320NEXT K1
33   1330LET J[5]=0
34   1340PRINT "ALL ENTRIES IN TEMFIL:DA COMPLETED."
35   1350CHAIN "DIRFIX:DA"
36   1360PRINT "NO REGULAR FILE TO SETUP."
37   1370STOP
38   1380LET J=J+DM
39   1390LET FM=J
40   1400IF J+D0<= D2RETURN
41   1410LET J=0
42   1420LET FM=1
43   1430RETURN
44   1440PRINT "DATA FILE ";N$;" NOT AVAILABLE."
45   1450GOTO 1490
46   1460PRINT "STATISTICS FILE ";N$;" NOT AVAILABLE."
47   1470GOTO 1490
48   1480PRINT "STATISTICS FILE ";N$;" INCOMPLETE."
49   1490PRINT J[1];" ENTRIES COMPLETED FROM TEMFIL:DA."
50   1500LET J[5]=1
51   1510CHAIN "DIRFIX:SV"
52   1520END
```

```
:5  T=00004 IS ON CR32767 USING 00012 BLKS K=0016

 1  10REM - PROGRAM DIRFIX:SV
 2  20REM - PROGRAM TO SET-UP SAMPLE FILES FROM TEMFIL:DA,
 3  30REM - THIS SEGMENT POSITIONS FILE POINTERS TO THE
 4  40REM - PROPER LOCATION IN STATUS:DA AND CREFT:DA.
 5  50REM - IT THEN ENTERS THE FILES IN STATUS AND CREFT.
 6  60REM - VERSION DATE 2/24/78
 7  70DIM J[3]
 8  80DIM A$[1/5],B$[24],C[100],D[2,5],C$[4],T$[10]
 9  90DIM A$[12],V$[22],Z$[252],A[64]
10  100FILES STATUS:DA,CREFT:DA,TEMFIL:DA
11  110LET K$=J[2]
12  120LET J3=0
13  130LET J1=1
14  140IF END #1 THEN 850
15  150READ #1,J1
16  160FOR J2=1TO 61STEP 4
17  170READ #1;A[J2],A[J2+1],A[J2+2],A[J2+3]
18  180IF A[J2]=-16010 220
19  190NEXT J2
20  200LET J1=J1+1
21  210GOTO 140
22  220LET K1=1
23  230IF END #2 THEN 830
24  240READ #2,K1;Z$
25  250FOR K2=1TO 241STEP 12
26  260IF Z$[K2,K2+5]="******"GOTO 300
27  270NEXT K2
28  280LET K1=K1+1
29  290GOTO 230
30  300FOR L1=1TO 6
31  310READ #3,K1
32  320FOR L2=1TO 10
33  330READ #3;V$
34  340IF V$[1,5]="       "GOTO 710
35  350IF J[2]=16010 360
36  360LET J[2]=J[2]-1
37  370GOTO 690
38  380IF J[1]=0GOTO 710
39  390LET J[1]=J[1]-1
40  400LET C$=V$[2,2]
41  410LET A[J2]=NUM(C$)-48
42  420LET C$=V$[1,1]
43  430LET A[J2]=A[J2]+NUM(C$)*100
44  440LET A1=0
45  450FOR I=3TO 6
46  460LET A1=A1*10
47  470LET C$=V$[I,I]
48  480LET A1=NUM(C$)-48+A1
49  490NEXT I
50  500LET A[J2+1]=A1
51  510LET C$=V$[10,10]
52  520LET N1=A[J2+3]=NUM(C$)-48
53  530LET A[J2+2]=D[2,N1]
54  540LET J2=J2+4
55  550IF J2<65GOTO 590
56  560GOSUB 690
57  570LET J1=J1+1
58  580LET J2=1
```

```
 9  590IF VS[7,10]="     "GOTO 690
 6  600LET Z$[K2,K2+5]=V$[7,12]
 1  610LET Z$[K2+6,K2+11]=VS[1,6]
 2  620LET K2=K2+12
 3  630IF K2<25 AGOTO 690
 4  640PRINT #2,K1;Z$
 5  650IF END #2 THEN 830
 6  660LET K1=K1+1
 7  670LET K2=1
 8  680LET J3=J3+1
 9  690NEXT L2
 0  700NEXT L1
 1  710LET A[J2]=A[J2+1]=-1
 2  720FOR J2=J2+2TO 64
 3  730LET A[J2]=0
 4  740NEXT J2
 5  750GOSUB 890
 6  760LET Z$[K2,K2+5]="******"
 7  770FOR K=K2+6TO 252
 8  780LET Z$[K,K]=" "
 9  790NEXT K
 0  800PRINT #2,K1;Z$
 1  810IF J[3]=0 CHAIN "WTMFIL:SV"
 2  820STOP
 3  830PRINT "CREFI:DA FULL!"
 4  840GOTO 860
 5  850PRINT "STATUS:DA FULL!";
 6  860PRINT "ONLY ";TAB(0);J3;" ENTRIES MADE";
 7  870PRINT " FROM TEMFIL:DA!"
 8  880STOP
 9  890PRINT #1,J1
 0  900FOR I=1TO 64
 1  910PRINT #1;A[I]
 2  920NEXT I
 3  930RETURN
 4  940END
```

DATA ENTRY PROGRAMS

```
10REM - PROGRAM ANALYZ.SV
20REM - PROGRAM FOR START OF ANALYSIS.
30REM - SETS STATUS FLAGS IN ANALYSIS AND STATUS FILES
40REM - TO STARTED.
50REM - THIS SECTION INITIALIZES METHODS INFORMATION.
60REM - ONLY CALLED ON LOADING.
70REM - 2/24/78 VERSION.
80DIM Z[2],B$[24],L$[5],L5[9],A$[175],I[2,5],J[100]
90DIM A[2],C[24],D$[9],Z$[240],H$[28]
100LET D$="DOSUM :DA"
110LET E$[1,9]="      :DA"
120LET C$[1,5]="     "
130FILES*,*
140ASSIGN "METHOD:DA",1,E
150IF E=2GOTO 180
160PRINT "METHOD:DA NOT AVAILABLE!"
170STOP
180READ #1;A$,D$
190READ #1,2
200FOR I=1TO 100
210READ #1;J[I]
220NEXT I
230FOR I=1TO 2
240FOR J=1TO 5
250READ #1;I[I,J]
260NEXT J
270NEXT I
280CHAIN "ANLY21:SV"
290END
```

```
G47 T=00004 IS ON CK32707 USING 00012 BLKS K=0016

 1  10REM - PROGRAM ANLYZ1:SV
 2  20REM - PROGRAM FOR START OF ANALYSIS.
 3  30REM - SETS STATUS FLAGS IN ANALYSIS AND STATUS FILES
 4  40REM - IS STARTED.
 5  50REM - 2/24/76 VERSION.
 6  60DIM Z[2],B$[24],C$[5],E$[9],A$[175],I[2,5],J[100]
 7  70DIM A[2],C[24],D$[9],Z$[240],H$[20]
 8  80FILES*,*
 9  90PRINT "ENTER SAMPLE TYPE: ";
10 100INPUT C$[1,4]
11 110IF C$[2,4]="   "STOP
12 120LET J=-5
13 130FOR I=1TO 5
14 140LET J=J+4
15 150IF C$[1,4]=B$[J,J+3]GOTO 190
16 160NEXT I
17 170PRINT "ILLEGAL SAMPLE TYPE!"
18 180GOTO 100
19 190LET Z[1]=1
20 200IF I=5GOTO 490
21 210LET Z[2]=I[2,I]
22 220CALLCHR$(46+I,D$[6,6])
23 230ASSIGN D$[1,9],1,E
24 240IF E=5GOTO 270
25 250PRINT D$;" NOT AVAILABLE!"
26 260GOTO 90
27 270PRINT "ENTER ANALYSIS NAME:";
28 280INPUT C$[1,5]
29 290LET I0=Z[1]
30 300LET L=I[1,I0]
31 310LET I2=I[2,I0]-1
32 320FOR J=0TO I2
33 330LET K=(J[J+L]-1)*5+1
34 340IF C$[1,5]=A$[K,K+4]GOTO 500
35 350NEXT J
36 360PRINT "UNDEFINED ANALYSIS FOR THIS SAMPLE TYPE!"
37 370GOTO 270
38 380LET J0=J*2+1
39 390READ #1,J0;S1
40 400IF S1<16GOTO 450
41 410IF S1<16GOTO 460
42 420PRINT "PRECISION ";
43 430PRINT "O.L. OUT OF CONTROL!"
44 440GOTO 500
45 450GOTO 240
46 460READ #1,J0+1;S1
47 470IF S1<16GOTO 500
48 480PRINT "ACCURACY ";
49 490GOTO 430
50 500PRINT "ENTER SAMPLE NO.";
51 510INPUT E$[1,5]
52 520IF E$[2,5]="     "GOTO 270
53 530ASSIGN E$[1,9],2,E
54 540IF E=5GOTO 560
55 550GOTO 500
56 560READ #2;A[1],A[2],H$
57 570LET I1=Z[1]*4-3
58 580IF D$[I1,I1+3]=H$[7,10]GOTO 610
```

```
590 PRINT "SAMPLE ";E$[1,6];" IS WRONG SAMPLE TYPE!"
600 GOTO 300
610 IF M$[11,14]="    "GOTO 950
620 REM - MUST BE A REAL SAMPLE.
630 READ #2;Z$[1,200]
640 LET P0=100
650 LET P[3]=0
660 LET P[2]=1
670 GOTO 730
680 LET E=TYP(-2)
690 IF E#0 GOTO 720
700 PRINT "ANALYSIS NOT IN FILE! TRY AGAIN!"
710 GOTO 300
720 READ #2,P[2];Z$[1,240]
730 LET I=P[3]+1
740 IF Z$[I,I+4]=C$[1,5]GOTO 810
750 LET P[3]=P[3]+40
760 IF P[3]<= P0 GOTO 730
770 LET P0=220
780 LET P[2]=P[2]+1
790 LET P[3]=0
800 GOTO 720
810 REM - FOUND ANALYSIS FILE
820 LET F$=Z$[I+37,I+37]
830 IF F$="1"GOTO 1540
840 IF F$="2"GOTO 1500
850 REM - SET STATUS TO "STARTED"
860 LET Z$[I+37,I+37]="1"
870 LET S1=S2=0
880 IF P[2]=1 GOTO 920
890 PRINT #2,P[2];Z$
900 READ #2,1;A[1],A[2],M$,Z$[1,200]
910 LET S1=1
920 GOSUB 1400
930 PRINT #2,1;A[1],A[2],M$,Z$[1,200]
940 CHAIN "ANLY22:SV"
950 LET S1=0
960 REM - START WITH RECORD 1
970 LET P[2]=1
980 LET I3=0
990 LET E1=E2=0
1000 GOTO 1050
1010 LET I3=8
1020 READ #1,P[2]
1030 LET I2=-2
1040 LET I1=-17
1050 FOR I=1 TO I3
1060 LET I1=I1+18
1070 LET I2=I2+3
1080 IF E#0 #2THEN 1210
1090 READ #2;Z$[I1,I1+17],C[I2],C[I2+1],C[I2+2]
1100 IF Z$[I1,I1+4]#C$[1,5]GOTO 1140
1110 IF Z$[I1,I1+3]="$$$$"GOTO 1260
1120 LET Z$[I1+17,I1+17]="1"
1130 LET E2=P[2]
1140 LET E=TYP(-2)
1150 IF E=3 GOTO 1200
1160 NEXT I
1170 IF E2>0 GOTO 1250
1180 LET P[2]=P[2]+1
```

```
1190GOTO 1010
1200LET E1=1
1210GOTO 1200
1220PRINT "FILE ERROR.  END FOUND BEFORE ANALYSIS BLOCK."
1230GOTO 500
1240REM - WRITE BACK WITH ADDED DATA.
1250LET E1=13
1260IF P(2)>1GOTO 1310
1270GOSUB 1450
1280PRINT #2,1;A(1),A(2),M$
1290LET I3=0
1300GOTO 1330
1310PRINT #1,P(2)
1320LET I3=0
1330IF E2#P(2)GOTO 1360
1340LET I3=E1
1350LET S1=1
1360LET I2=-2
1370LET I1=-17
1380FOR I=1TO 13
1390LET I1=I1+16
1400LET I2=I2+3
1410PRINT #2;Z$(I1,I1+17),C(I2),C(I2+1),C(I2+2)
1420NEXT I
1430IF S1=1GOTO 1400
1440LET P(2)=E2
1450GOTO 1010
1460LET S1=0
1470CHAIN "ANLYZ2.SV"
1480IF M$(27,28)#"00"RETURN
1490LET K1=INT(Z(2)/10)
1500LET K2=Z(2)-K1*10
1510CALL CHRS(K1+48,M$(27,27))
1520CALL CHRS(K2+48,M$(28,28))
1530RETURN
1540PRINT "ALREADY STARTED!"
1550GOTO 500
1560PRINT "DONE!"
1570GOTO 500
1580END
```

```
PROG48 T=00004 IS ON CR32707 USING 00012 BLKS R=0016

0001   10REM - PROGRAM ANLYZ2:SV
0002   20REM - SECOND PART OF ANALYZ.  CHECKS AND SETS
0003   30REM - STATUS FILE AND THEN CHAINS BACK TO ANALYZ.
0004   40REM - 2/24/78 VERSION
0005   50COM Z(2),B$(24),C$(5),E$(9),A$(175),I(2,5),J(100)
0006   60COM A(2),C(200),D$(9),Z$(2400),H$(20)
0007   70DIM B(64)
0008   80FILE#1STATUS:DA
0009   90LET J1=0
0010   100LET J=1
0011   110IF END #1THEN 320
0012   120READ #1,J
0013   130FOR I=1TO 64
0014   140READ #1;B(I)
0015   150NEXT I
0016   160FOR I=1TO 61STEP 4
0017   170IF I=1AND J=1GOTO 200
0018   180IF A(1)#B(I)GOTO 240
0019   190IF A(2)=B(I+1)GOTO 230
0020   200NEXT I
0021   210LET J=J+1
0022   220GOTO 110
0023   230IF B(I+3)>20GOTO 260
0024   240LET B(I+3)=100
0025   250GOTO 270
0026   260LET B(I+3)=B(I+3)+100
0027   270PRINT #1,J
0028   280FOR I=1TO 64
0029   290PRINT #1;B(I)
0030   300NEXT I
0031   310CHAIN "ANLYZ1:SV",500
0032   320END
```

UTILITY AND MAINTENANCE PROGRAMS

```
**** T=00000 IS ON LU 05

 11  10REM - DETECTION LIMIT AND ERROR PROGRAM.
 12  20REM - USES METHOD OF HUBAUX AND VOS.
 13  30REM - 12/22/77 REV.
 14  40DIM W[31],X[60],Y[60],C[5],L5[5]
 15  50DATA 80,90,95,98,99
 16  60FOR I=1TO 5
 17  70READ C[I]
 18  80NEXT I
 19  90LET L=1
 20  100LET Z3=0
 21  110PRINT "DETECTION LIMIT PROGRAM"
 22  120PRINT "LEGAL COMMANDS ARE:"
 23  130PRINT "  E=EXIT"
 24  140PRINT "  R=RESTART"
 25  150PRINT "  LN=N REPLICATES"
 26  160PRINT "  SN=N STANDARDS"
 27  170PRINT "  CMN=MN CONFIDENCE LEVEL"
 28  180PRINT "WHAT IS THE NUMBER OF INDEPENDENT CALIBRATION VALUES?"
 29  190FILES TTABLE:DA
 30  200INPUT N
 31  210IF N<= 2THEN 750
 32  220LET I=T=0
 33  230PRINT "ENTER THE CONFIDENCE LEVEL DESIRED"
 34  240GOSUB 810
 35  250INPUT Z1
 36  260GOSUB 1230
 37  270IF Z1#0GOTO 300
 38  280GOSUB 800
 39  290GOTO 260
 40  300PRINT "ENTER THE NUMBER OF INPUT VALUES";
 41  310INPUT W
 42  320IF W<NTHEN 830
 43  330PRINT "ENTER X,Y VALUES AS X-Y PAIRS."
 44  340FOR I=1TO W
 45  350INPUT X[I],Y[I]
 46  360NEXT I
 47  370LET A1=A2=C1=C2=C3=0
 48  380FOR I=1TO W
 49  390LET A1=A1+X[I]
 50  400LET C1=C1+Y[I]
 51  410LET A2=A2+X[I]*X[I]
 52  420LET C2=C2+X[I]*Y[I]
 53  430LET C3=C3+Y[I]*Y[I]
 54  440NEXT I
 55  450LET X1=A2-A1*A1/W
 56  460LET Y1=C3-C1*C1/W
 57  470LET Z1=C2-C1*A1/W
 58  480LET M1=Z1/X1
 59  490LET B1=(C1*A2-A1*C2)/(W*X1)
 60  500PRINT "THE SLOPE IS ";TAB(0);M1
 61  510PRINT "THE INTERCEPT IS ";TAB(0);B1
 62  520LET S1=SQR((Y1-Z1T2/X1)/(N-2))
 63  530PRINT "THE STANDARD DEVIATION IS "S1
 64  540LET E1=A1/W
 65  550PRINT "ENTER COMMAND:";
 66  560INPUT L$
 67  570LET A$=L$[2,2]
 68  580GOTO 880
```

```
590LET S=S1*T
600IF S1=0GOTO 690
610LET K1=B1+S*SQR(1+1/N+E1↑2/X1)
620LET C0=M1/S
630LET D=(K1-B1)/S
640LET A=1-X1*C0↑2
650LET B=2*(C0*D*X1-E1)
660LET C=E1↑2+X1*(1/L+1/N-D↑2)
670LET L1=(-B-SQR(B↑2-4*A*C))/(2*A)
680GOTO 700
690LET K1=L1=0
700PRINT "FOR ";TAB(0);N;TAB(0);" STANDARDS WITH ";
710PRINT TAB(0);L;TAB(0);" SAMPLE REPLICATES:"
720PRINT " THE DECISION LIMIT IS ";TAB(0);K1
730PRINT " THE DETECTION LIMIT IS ";TAB(0);L1
740GOTO 550
750PRINT "REENTER A VALUE >2 AS PROGRAM BASED ON ";
760PRINT "TWO DEGREES OF FREEDOM."
770GOTO 180
780PRINT "REENTER A POSITIVE VALUE"
790GOTO 550
800PRINT "REENTER THE CONFIDENCE LEVEL."
810PRINT "50,90,95,99 OR 99% ";
820RETURN
830PRINT "THE TOTAL NUMBER OF X,Y INPUT VALUES MUST AT LEAST";
840PRINT "EQUAL THE NUMBER OF CALIBRATION VALUES."
850PRINT " REENTER THE TOTAL NUMBER."
860GOTO 510
870STOP
880IF L$(1,1)="E"THEN 970
890IF L$(1,1)="R"GOTO 180
900IF L$(1,1)="C"GOTO 1010
910IF L$(1,1)#"S"THEN 970
920IF A$=" "GOTO 1190
930GOSUB 1080
940LET N=X
950GOTO 590
960GOSUB 1400
970IF L$(1,1)#"L"THEN 1190
980GOSUB 1080
990LET L=X
1000GOTO 590
1010GOSUB 1080
1020LET Z1=X
1030GOSUB 1230
1040IF Z1=0GOTO 1060
1050GOTO 590
1060GOSUB 600
1070GOTO 1030
1080GOSUB 1100
1090IF X<0GOTO 1120
1100IF LEN(L$)=2RETURN
1110LET L0=X
1120LET A$=L$[3,3]
1130GOSUB 1100
1140LET X=X+L0*10
1150RETURN
1100LET X=NUM(A$)-48
1170IF X=-16RETURN
1180IF X>=0GOTO 1210
```

```
1190PRINT "UNRECOGNIZED COMMAND!"
1200GOTO 550
1210IF X>9THEN 1190
1220RETURN
1230FOR Z=1TO 5
1240IF C[Z]=Z1THEN 1280
1250NEXT Z
1260LET Z1=0
1270RETURN
1280IF Z3=ZTHEN 1420
1290LET Z1=INT((Z+1)/2)
1300LET Z3=Z
1310LET Z2=Z+1-2*Z1
1320READ #1,Z1
1330IF Z2=0THEN 1370
1340FOR K=1TO 31
1350READ #1;Z1
1360NEXT K
1370FOR K=1TO 31
1380READ #1;W[K]
1390NEXT K
1400LET K=N
1410IF K>31LET K=31
1420LET T=W[K]
1430PRINT "THE VALUE OF T IS ";TAB(1);T
1440RETURN
1450END
```

```
G30 T=00004 IS ON CR32767 USING 00012 BLKS R=0016

 1  10REM - PROGRAM #TMFIL:SV
 2  20REM - PROGRAM TO CREATE TEMPORARY FILE OF
 3  30REM - INFORMATION FOR USE IN WRITING OUT
 4  40REM - ANALYSIS FILES.
 5  45REM - VERSION DATA 2/8/76.
 6  50REM - IS TEMFIL:DA IS 6 BLOCKS LONG.
 7  60DIM Z$[22]
 8  70LET Z$="                      "
 9  80FILES*
 0  90ASSIGN "TEMFIL:DA",1,E
 1  100IF E=0THEN 130
 2  110PRINT "TEMFIL:DA NOT AVAILABLE!"
 3  120STOP
 4  130GOSUB 150
 5  140GOTO 190
 6  150FOR I=1TO 10
 7  160PRINT #1;Z$
 8  170NEXT I
 9  180RETURN
 0  190PRINT #1;1,0,00,00
 1  200FOR J=210 6
 2  210PRINT #1,J
 3  220GOSUB 150
 4  230NEXT J
 5  240PRINT #1;END
 6  250STOP
 7  260PRINT #1;0,END
 8  270STOP
 9  280END
```

```
JG31 T=00004 IS ON CR32/07 USING 00012 BLKS R=0016

 10REM - PROGRAM #STATS:SV
 20REM - PROGRAM TO CREATE NEW STATUS FILE OR CHANGE ALPHABETIC
 30REM - PREFIX DESIGNATOR FOR SAMPLE FILE NAMES -
 35REM - VERSION DATA 1/30/78.
 40DIM A[64],N$[6]
 50FILESSTATUS:DA
 60LET N2=60
 70LET N1=5
 75LET E2=0
 80LET E=TYP(-1)
 90IF E=3THEN 500
100READ #1;A[1],A[2],A[3],A[4]
110LET E1=0
110LET E2=0
120REM - READ REST OF FIRST BLOCK
130GOSUB 210
140REM - NOW READ NEW SAMPLE PREFIX
150GOSUB 300
160REM - NOW REWRITE THE BLOCK, FIRST SETTING BLOCK POINTER
170PRINT #1,1
180GOSUB 300
190IF E1=1THEN 520
200STOP
210REM - SUBROUTINE TO READ BALANCE OF BLOCK
220FOR N=N1TO N2STEP 4
230LET E=TYP(-1)
240IF E=3GOTO 270
250READ #1;A[N],A[N+1],A[N+2],A[N+3]
260NEXT N
270RETURN
275LET N2=N-1
280LET E1=1
290RETURN
300REM - SUBROUTINE TO WRITE OUT BLOCK.
310FOR N=110 N2STEP 4
320PRINT #1;A[N],A[N+1],A[N+2],A[N+3]
330NEXT N
340RETURN
350REM - SUBROUTINE TO INPUT NEW SAMPLE PREFIX.
360PRINT "ENTER SAMPLE NO. PREFIX ";
370INPUT A$
380LET A[1]=NUM(A$)*100
390LET A[2]=1
400IF E2=0RETURN
410PRINT "ENTER NEXT SAMPLE NUMBER:";
420INPUT N$[1,5]
430IF N$[2,5]="    "RETURN
435LET A$=N$[1,1]
436LET A[1]=A[1]+NUM(A$)-48
440LET A=0
450FOR I=2TO 5
460LET A$=N$[I,I]
470LET A=A*10+NUM(A$)-48
475NEXT I
480LET A[2]=A
490RETURN
500REM - READ PREFIX
510GOSUB 350
```

```
 59  510PRINT #1;A[1],A[2],0,0
 60  520PRINT #1;END
 61  530STOP
 62  540END
```

```
G32 T=00004 IS ON CR32767 USING 00012 BLKS K=0016

 1   10REM - PROGRAM WQCTIM:SV
 2   20REM - PROGRAM TO CREATE QCTIME FILE.
 3   30REM - QCTIME:DA REQUIRES 5 RECORDS.
 4   35REM - VERSION DATE 1/30//8.
 5   40FILES QCTIME:DA
 6   50FOR I=1TO 5
 7   60PRINT #1,I;I
 8   70PRINT #1;0,0,0,3,0,0
 9   80FOR J=1TO 5
 0   90PRINT #1,0,0,0,0,0,0,0,0,0
 1   100NEXT J
 2   110PRINT #1;0,0,0,0,0,0,0,0
 3   120NEXT I
 4   130PRINT #1;END
 5   140END
```

```
UG33 T=00004 IS ON CR32707 USING 00012 BLKS K=0016

01   10REM - PROGRAM KGRSIZ:SV
02   20REM - WRITE-OUT GROUP SAMPLE TABLE.
03   30REM - THIS VERSION ASSUMES THAT THE ORDER OF
04   40REM - STANDARD CONCENTRATIONS IN ALL PROGRAMS IS
05   50REM - .5,2,10,50,1,5,20
06   60REM - VERSION DATE 2/8/76.
07   70FILES GRSIZE:DA
08   80DIM A[5,5]
09   90REM - GROUP SIZE
10   100DATA 21,10,11,0,1
11   110REM - BLANKS ( ZERO CONC. STANDARD )
12   120DATA 4,3,3,2,1
13   130REM - DUPLICATES.
14   140DATA 4,3,3,2,1
15   150REM - STANDARDS
16   160DATA 7,6,5,4,3
17   170REM - SPIKES
18   180DATA 7,6,5,4,3
19   190REM - STANDARDS
20   200DATA 7,6,5,4,3
21   210FOR I=1TO 5
22   220FOR J=1TO 5
23   230READ A[I,J]
24   240NEXT J
25   250NEXT I
26   260FOR J=1TO 5
27   270FOR I=1TO 5
28   280PRINT #1;A[I,J]
29   290NEXT I
30   300NEXT J
31   310PRINT #1;END
32   320STOP
33   330END
```

```
)G34 I=000004 IS ON CR32767 USING 00012 BLKS R=0016

)1   10REM - PROGRAM WTTABL:SV
)2   20REM - PROGRAM TO WRITE FILE OF STUDENTS "T"
)3   25REM - FOR MANUAL ENTRY VERSION OF HUBBAUX-VOS.
)4   28REM - VERSION DATA 1/10/78
)5   30FILEST[ABLE:DA
)6   40REM "T" FOR 69%
)7   50DATA 3.078,1.800,1.638,1.533,1.476,1.44
)8   60DATA 1.415,1.397,1.383,1.372,1.363,1.356
)9   70DATA 1.35,1.345,1.341,1.337,1.333,1.33
10   80DATA 1.328,1.325,1.323,1.321,1.319,1.318
11   90DATA 1.316,1.315,1.314,1.313,1.311,1.31,1.282
12   100REM "T" FOR 90%
13   110DATA 6.314,2.92,2.353,2.132,2.015,1.943
14   120DATA 1.895,1.86,1.833,1.812,1.795,1.782
15   130DATA 1.771,1.761,1.753,1.746,1.74,1.734
16   140DATA 1.729,1.725,1.721,1.717,1.714,1.711
17   150DATA 1.708,1.706,1.703,1.701,1.699,1.697,1.645
18   160REM "T" FOR 95%
19   170DATA 12.706,4.303,3.182,2.776,2.571,2.447
20   180DATA 2.365,2.306,2.262,2.228,2.201,2.179
21   190DATA 2.16,2.145,2.131,2.12,2.11,2.101
22   200DATA 2.093,2.086,2.08,2.074,2.069,2.064
23   210DATA 2.06,2.056,2.052,2.048,2.045,2.042,1.96
24   220REM "T" FOR 98%
25   230DATA 31.821,6.965,4.541,3.747,3.365,3.143
26   240DATA 2.998,2.896,2.821,2.764,2.718,2.681
27   250DATA 2.65,2.624,2.602,2.583,2.567,2.552
28   260DATA 2.539,2.528,2.518,2.508,2.5,2.492
29   270DATA 2.485,2.479,2.473,2.467,2.462,2.457,2.326
30   280REM "T" FOR 99%
31   290DATA 63.657,9.925,5.841,4.604,4.032,3.707
32   300DATA 3.49,3.355,3.25,3.169,3.106,3.055
33   310DATA 3.012,2.977,2.947,2.921,2.898,2.878
34   320DATA 2.861,2.845,2.831,2.819,2.807,2.797
35   330DATA 2.787,2.779,2.771,2.763,2.756,2.75,2.576
36   340REM - END OF "T" DATA
37   350DIM N[62]
38   360FOR I=1TO 2
39   370FOR J=1TO 62
40   380READ N[J]
41   390NEXT J
42   430PRINT #1,I;N[1]
43   410FOR J=2TO 62
44   420PRINT #1;N[J]
45   430NEXT J
46   440NEXT I
47   450FOR J=1TO 31
48   460READ N[J]
49   470NEXT J
50   480PRINT #1,3;N[1]
51   490FOR J=2TO 31
52   500PRINT #1;N[J]
53   510NEXT J
```

```
PROG35 T=00004 IS ON CR32767 USING 00012 BLKS R=0016

0001   10REM - PROGRAM TAB90T:SV
0002   20REM - PROGRAM TO WRITE FILE OF STUDENTS "T"
0003   25REM - THIS VERSION 90% VALUES ONLY.
0004   30FILESTTAB90:DA
0005   100REM "T" FOR 90%
0006   110DATA 6.314,2.92,2.353,2.132,2.015,1.943
0007   120DATA 1.895,1.86,1.833,1.812,1.795,1.782
0008   130DATA 1.771,1.761,1.753,1.746,1.74,1.734
0009   140DATA 1.729,1.725,1.721,1.717,1.714,1.711
0010   150DATA 1.708,1.706,1.703,1.701,1.699,1.697,1.645
0011   160FOR I=1TO 31
0012   170READ N
0013   180PRINT #1;N
0014   190NEXT I
0015   200PRINT #1;END
0016   210END
```

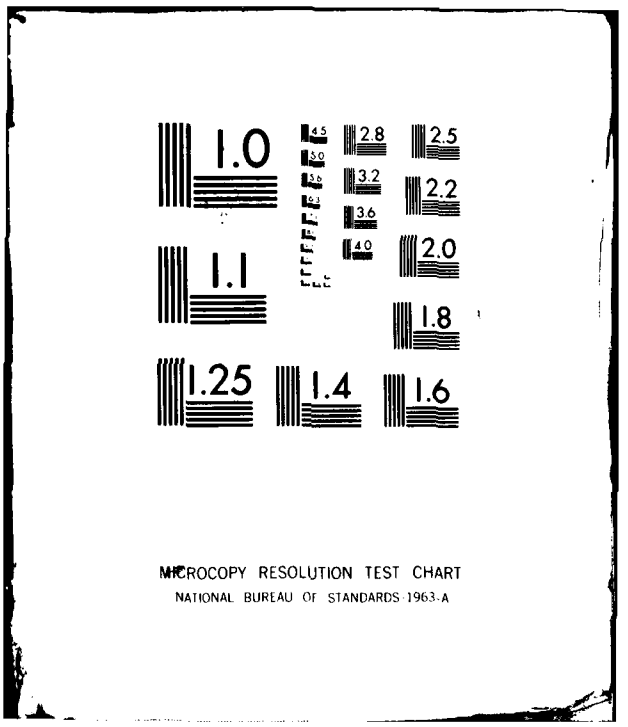
```
)G36 T=00004 IS ON CR32707 USING 00012 BLKS R=0016

)1   10REM - PROGRAM WMETHD:SV
)2   20REM - PROGRAM TO SET UP FILE OF METHODS INFORMATION
)3   25REM - VERSION DATA 12/27/77
)4   30DIM A$(175),B$(24),C(100),D(2,5)
)5   40FILE$METHOD:DA
)6   50REM - ANALYTE ABBREVIATIONS.
)7   60LET A$(1,45)="CL   F    NIT  PH   SO4  HARD NA   DIMP DCPD "
)8   70LET A$(46,90)="S    SUXIDSUNE DXAT DITH ALDRNDLDRNENDRN"
)9   80LET A$(91,135)="DDE  P4   CU   CUAEXAS   ASAEXHG   HGAEX"
10   90LET A$(136,175)="K    CA   COND DMMP DMP  FE   OPO4 P4   "
11   100REM - ANALYTES FOR EACH TYPE OF ANALYSIS.
12   110LET B$(1,24)="CPH CPS CPW 300 BSNFSPEC"
13   120REM - COMP PILOT BIOL.
14   130DATA 8,10,11,12,13,14,15,16,17,18,19,20,21,23,25,27
15   140REM - COMP PILOT SOIL.
16   150DATA 1,4,5,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25
17   160DATA 26,27,28,29,30
18   170REM - COMP PILOT WATER.
19   180DATA 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21
20   190DATA 23,25,27,28,29,30
21   [illegible]REM - 300
22   210DATA 1,2,3,4,5,6,7,8,9
23   220REM - BASIN F.
24   230DATA 1,3,4,5,6,8,9,13,14,15,16,17,18,19,20,21,31,32,33,34,35
25   240REM - ARRAY START AND SIZE
26   250DATA 1,17,44,71,88,16,27,27,9,21
27   260FOR I=1TO 100
28   270READ C(I)
29   280NEXT I
30   290FOR I=1TO 2
31   300FOR J=1TO 5
32   310READ D(I,J)
33   320NEXT J
34   330NEXT I
35   340PRINT #1;A$,B$
36   350PRINT #1,2
37   360FOR I=1TO 100
38   370PRINT #1;C(I)
39   380NEXT I
40   390FOR I=1TO 2
41   400FOR J=1TO 5
42   410PRINT #1;D(I,J)
43   420NEXT J
44   430NEXT I
45   440REM - START NEW RECORD.
46   450PRINT #1,4
47   460REM - GC ANALYSES
48   470DATA 8,9,15,16,17,18,19,20,31,32
49   480REM - AA ANALYSES
50   490DATA 7,21,22,23,24,25,26,27,28,29,33
51   500REM - TC ANALYSES
52   510DATA 1,2,3,4,5,6,10,11,12,13,14,30,34,35
53   520FOR I=1TO 35
54   530READ C(I)
55   540NEXT I
56   541FOR I=1TO 35
57   542PRINT #1;C(I)
58   543NEXT I
```

AD-A084 896 BATTELLE COLUMBUS LABS OHIO TACTICAL TECHNOLOGY CENTER F/G 9/2
A QUALITY CONTROL DATA MANAGEMENT SYSTEM.(U)
FEB 78 M B NEHER, R E HEFFELFINGER DAAK40-73-C-0142
UNCLASSIFIED BATT-CDIR-1-2 DRXTH -TD-CR-80069 NL

2 - 2
AD
A084896

END
DATE FILMED
7 80
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
0059  550REM - ARRAY START AND SIZE
0060  560DATA 1,11,22,10,11,14
0061  570REM - FILE RECORD LENGTHS.
0062  580DATA 3,5,5,2,4,3,4,4,2,3
0063  590LET K=1
0064  600FOR L=1TO 2
0065  610LET K=K+2
0066  620FOR I=1TO 2
0067  630FOR J=1TO K
0068  640READ D[I,J]
0069  650NEXT J
0070  660NEXT I
0071  670FOR I=1TO 2
0072  680FOR J=1TO K
0073  690PRINT #1;D[I,J]
0074  700NEXT J
0075  710NEXT I
0076  720NEXT L
0077  730PRINT #1;END
0078  740END
```

```
ROG37 T=00004 IS ON CK32767 USING 00012 BLKS R=0016

001   10REM - PROGRAM FILCHK:SV
002   12REM - PROGRAM TO READ AND PRINT SAMPLE FILES.
003   15REM - VERSION DATA 2/8/78
004   20DIM Z$[240],A$[9],H$[26]
005   30LET J=1
006   40FILES*
007   50LET K0=0
008   60LET D0=J=1
009   70PRINT "ENTER SAMPLE NUMBER";
010   80INPUT A$[1,6]
011   90IF A$[2,6]="      "STOP
012   100ASSIGN A$,1,E
013   110IF E#0GOTO 430
014   120READ #1;A0,A1,H$
015   130PRINT A0;A1;H$
016   140IF H$[11,15]="      "GOTO 270
017   150REM - USER SAMPLE BRANCH
018   160READ #1;Z$[1,200]
019   170LET D1=40
020   180LET D2=200
021   190GOTO 230
022   200GOSUB 350
023   210IF END #1THEN 450
024   220IF F0=1READ #1;Z$
025   230LET F0=0
026   240IF Z$[D0,D0+3]="$$$$"GOTO 450
027   250PRINT Z$[D0,D0+39]
028   260GOTO 200
029   270REM =U.C.SAMPLE BRANCH.
030   280LET D1=29
031   290LET D2=240
032   300GOSUB 350
033   310LET F0=0
034   320READ #1;Z$[1,18],A1,A2,A3
035   330PRINT Z$[1,18];A1;A2;A3
036   340GOTO 300
037   350LET D0=D0+D1
038   360LET F0=0
039   370IF D0<D2RETURN
040   380LET D0=1
041   390LET J=J+1
042   400IF END #1THEN 450
043   410LET F0=1
044   420RETURN
045   430PRINT A$;" NOT AVAILABLE!"
046   440GOTO 50
047   450PRINT "END OF SAMPLE."
048   460PRINT
049   470PRINT
050   480GOTO 50
051   490END
```

```
)G20 T=00004 IS ON CK32767 USING 00012 BLKS K=0016

)1   10REM - PROGRAM CREF:SV
)2   20REM -UTILITY PROGRAM TO SEARCH CROSS-REFERENCE
)3   30REM - FOR USER SAMPLE NO. AND REPORT STATUS.
)4   40REM - REVISION DATE 2/24/76.
)5   50DIM N$[20],Z$[240],B$[9],A[6],G$[5],J$[1]
)6   60DIM A$[252],U$[6],C$[24],B[2],L[64],D$[4]
)7   70LET G$="LPT CPS CPM 360 DSNFSPEC"
)8   80LET D$[1,9]="      :DA"
)9   90FILES CREFT:DA,STATUS:DA,*
)0   100REM - ENTRY FOR MANUAL SEARCH
)1   110PRINT "ENTER USER NO.: ";
)2   120INPUT U$[1,6]
)3   130IF U$[2,6]="     "GOTO 250
)4   140LET J=1
)5   150IF END #1THEN 230
)6   160READ #1,J;A$
)7   170FOR I=1TO 241STEP 12
)8   180IF A$[I,I+5]=U$GOTO 310
)9   190IF A$[I,I+5]="******"GOTO 230
)0   200NEXT I
)1   210LET J=J+1
)2   220GOTO 150
)3   230PRINT "USER NO. NOT IN CROSS REFERENCE FILE!"
)4   240GOTO 110
)5   250LET I=1
)6   260PRINT "ENTER LAB. NO.";
)7   270LET A$[1,6]="      "
)8   280INPUT A$[7,12]
)9   290IF A$[8,12]="     "STOP
)0   300GOTO 320
)1   310PRINT "LAB. NO. IS ";A$[I+6,I+11]
)2   320LET I0=1
)3   330LET D$=A$[I+6,I+6]
)4   340LET B[1]=NUM(D$)*100
)5   350LET D$=A$[I+7,I+7]
)6   360LET B[1]=B[1]+NUM(D$)-48
)7   370LET B[2]=0
)8   380FOR K=8TO 11
)9   390LET D$=A$[I+K,I+K]
)0   400LET B[2]=B[2]*10+NUM(D$)-48
)1   410NEXT K
)2   420LET J=1
)3   430IF END #2THEN 540
)4   440READ #2,J
)5   450FOR I=1TO 16
)6   460READ #2;A[1],A[2],A[3],A[4]
)7   470IF I=1AND J=1GOTO 510
)8   480IF A[1]#B[1]GOTO 510
)9   490IF A[2]#B[2]GOTO 510
)0   500IF A[3]>0GOTO 560
)1   510NEXT I
)2   520LET J=J+1
)3   530GOTO 430
)4   540PRINT "LAB. NO. NOT IN STATUS FILE!"
)5   550GOTO 110
)6   560REM - FOUND LAB NO."
)7   570IF A[4]>10GOTO 610
)8   580LET C=(A[4]-1)*4+1
```

```
590PRINT "ANALYSIS TYPE ";C$[C,C+3];" NOT YET STARTED."
600GOTO 110
610LET A1=INT(A[4]/100)
620LET A2=A[4]-A1*100
630PRINT A1;" ANALYSES STARTED ";
640IF A2>0GOTO 670
650PRINT "NONE COMPLETED."
660GOTO 680
670PRINT TAB(0);A2;" COMPLETED."
680PRINT "DO YOU WANT STATUS OF INDIVIDUAL ANALYSES";
690INPUT D$[1,4]
700IF D$[1,1]="Y"GOTO 730
710IF D$[1,1]="N"GOTO 110
720PRINT "PLEASE ANSWER Y(ES) OR N(O)!"
730LET B$[1,6]=A$[10+6,10+11]
740ASSIGN B$[1,9],3,E
750ASSIGN B$,3,E
760IF E#0GOTO 1130
770LET D0=1
780READ #3;A0,A1,H$
790IF H$[11,15]="     "GOTO 950
800REM - USER SAMPLE BRANCH
810READ #3;Z$[1,200]
820LET D0=1
830LET D1=40
840LET D2=200
850GOTO 890
860GOSUB 1050
870IF END #3THEN 1150
880IF F0=1READ #3;Z$
890LET F0=0
900IF Z$[D0,D0+3]="$$$$"GOTO 110
910LET G$=Z$[D0,D0+4]
920LET J$=Z$[D0+37,D0+37]
930GOSUB 1180
940GOTO 860
950REM -U.C.SAMPLE BRANCH.
960LET D1=29
970LET D2=240
980GOSUB 1050
990LET F0=0
1000READ #3;Z$[1,10],A1,A2,A3
1010LET G$=Z$[1,5]
1020LET J$=Z$[10,10]
1030GOSUB 1160
1040GOTO 960
1050LET D0=D0+D1
1060LET F0=0
1070IF D0<D2RETURN
1080LET D0=1
1090LET J=J+1
1100IF END #3THEN 110
1110LET F0=1
1120RETURN
1130PRINT B$;" NOT AVAILABLE!"
1140GOTO 110
1150PRINT
1160PRINT
1170GOTO 110
1180PRINT G$;
```

```
19  11901F Js="0"PRINT " NOT STARTED."
20  12001F Js=" "PRINT " NOT STARTED."
21  12101F Js="1"PRINT " STARTED."
22  12201F Js="2"PRINT " COMPLETED."
23  12300RETURN
24  12400END
```

```
AUG22 T=00004 IS ON CR3270 USING 00012 BLKS K=0016

 100  10REM - PROGRAM CKCSUM:SV
 200  20DIM B$[24],C$[5],E$[9],A$[175],I[2,5],J[100]
 300  30DIM F$[4]
 400  40DIM D[5],L[5],O[61],X$[61],D$[9],M[5],Y[13],Y$[48]
 500  50LET Y$[1,48]=" JAN FEB MAR APR MAYJUNEJULY AUGSEPT OCT NOV DEC"
 600  60DATA 31,28,31,30,31,30,31,31,30,31,30,31,28
 700  70FOR I=1TO 13
 800  80READ Y[I]
 900  90NEXT I
1000  100LET Y1=TIM(3)
1100  110IF (INT(Y1/400))*400=Y1GOTO 130
1200  120IF (INT(Y1/100))*100=Y1LET Y[2]=Y[13]+1
1300  130LET D$="CUSUM :DA"
1400  140LET E$="       :DA"
1500  150FILES*
1600  160ASSIGN "METHOD:DA",1,E
1700  170IF E=0GOTO 200
1800  180PRINT "METHOD:DA NOT AVAILABLE!"
1900  190STOP
2000  200READ #1;A$,B$
2100  210READ #1,2
2200  220FOR I=1TO 100
2300  230READ #1;J[I]
2400  240NEXT I
2500  250FOR I=1TO 2
2600  260FOR J=1TO 5
2700  270READ #1;I[I,J]
2800  280NEXT J
2900  290NEXT I
3000  300PRINT "ENTER SAMPLE TYPE: ";
3100  310INPUT F$[1,4]
3200  320IF F$[2,4]="   "STOP
3300  330LET J=-5
3400  340FOR I=1TO 5
3500  350LET J=J+4
3600  360IF F$=B$[J,J+3]GOTO 400
3700  370NEXT I
3800  380PRINT "ILLEGAL SAMPLE TYPE!"
3900  390GOTO 310
4000  400IF I=0GOTO 650
4100  410LET K0=I[2,I]
4200  420CALLCHR$(48+I,D$[0,0])
4300  430ASSIGN D$,1,E
4400  440IF E=0THEN 470
4500  450PRINT D$," NOT AVAILABLE!"
4600  460GOTO 300
4700  470PRINT "ENTER ANALYSIS NAME:";
4800  480INPUT C$[1,5]
4900  490IF C$[2,5]="    "GOTO 300
5000  500LET L=I[1,I]
5100  510LET I2=I[2,I]-1
5200  520FOR J=0TO I2
5300  530LET K=(J[J+L]-1)*5+1
5400  540IF C$=A$[K,K+4]THEN 560
5500  550NEXT J
5600  560PRINT "UNDEFINED ANALYSIS FOR THIS SAMPLE TYPE!"
5700  570GOTO 470
5800  580LET J0=(J+1)*2
```

```
590PRINT "WHICH PART";
600INPUT G$
610IF G$[1,1]="P"GOTO 660
620IF G$[1,1]="A"GOTO 680
630IF G$[1,1]="E"GOTO 470
640PRINT "ENTER P FOR PRECISION, A FOR ACCURACY."
650GOTO 590
660LET J=J0-1
670GOTO 690
680LET J=J0
690READ #1,J;S1,L1,L2,L3,M1
700LET M0=1
710IF M1<58GOTO 760
720LET M0=M1-57
730IF M1>0GOTO 760
740LET I=1
750GOTO 790
760FOR I=1TO M1
770READ #1;D[I]
780NEXT I
790FOR K=ITO 61
800LET D[K]=0
810NEXT K
820REM - CALCULATE 3 POINTS ON UPPER AND MID LINES.
830FOR I=1TO 5
840LET L0=(I*15-15)*L3*M0
850LET U[I]=L1+L0
860LET M[I]=L0
870LET L[I]=L2+L0
880NEXT I
890LET F1=50/(U[5]-L[1])
900LET L0=ABS(INT(L[1]*F1+.5))+2
910FOR I=1TO 5
920LET U[I]=INT(U[I]*F1+.5)+L0
930LET L[I]=INT(L[I]*F1+.5)+L0
940LET M[I]=INT(M[I]*F1+.5)+L0
950NEXT I
960IF M1=0GOTO 1000
970FOR I=1TO 61
980LET D[I]=D[I]*F1
990NEXT I
1000GOSUB 1270
1010FOR I=51TO 1STEP -1
1020FOR K=1TO 81
1030LET X$[K]=" "
1040NEXT K
1050FOR L=1TO 5
1060LET N=1+(L-1)*15
1070IF U[L]=ILET X$[N,N]="+"
1080IF M[L]=ILET X$[N,N]="0"
1090IF L[L]=ILET X$[N,N]="-"
1100NEXT L
1110IF M1=0GOTO 1150
1120FOR N=1TO M1
1130IF D[N]=ILET X$[N]="*"
1140NEXT N
1150PRINT "I";X$;"I"
1160NEXT I
1170FOR I=1TO 81
1180LET X$[I]="-"
```

```
 1190 NEXT I
 1200 PRINT "I";X$;"I"
 1210 FOR I=1 TO 7
 1220 PRINT
 1230 NEXT I
 1240 GOTO 590
 1250 END
 1260 LET M1=(M1-1)*5
 1270 LET D1=TIM(2)
 1280 FOR I=1 TO 11
 1290 IF D1<Y(I+1) GOTO 1320
 1300 LET K=1
 1310 LET D1=D1-Y(I)
 1320 NEXT I
 1330 FOR I=1 TO 10
 1340 PRINT
 1350 NEXT I
 1360 LET K=(K-1)*4+1
 1370 IF G$(1,1)="P" GOTO 1400
 1380 PRINT "ACCURACY ";
 1390 GOTO 1410
 1400 PRINT "PRECISION ";
 1410 PRINT "CUSUM FILE FOR ";C$(1,5);" IN ";F$(1,5)
 1420 PRINT "   ";Y$(K,K+3);" ";D1;TAB(0);",";Y1
 1430 PRINT
 1440 RETURN
 1450 END
```

APPENDIX C

LOGIC FLOWCHARTS FOR MAJOR PROGRAMS

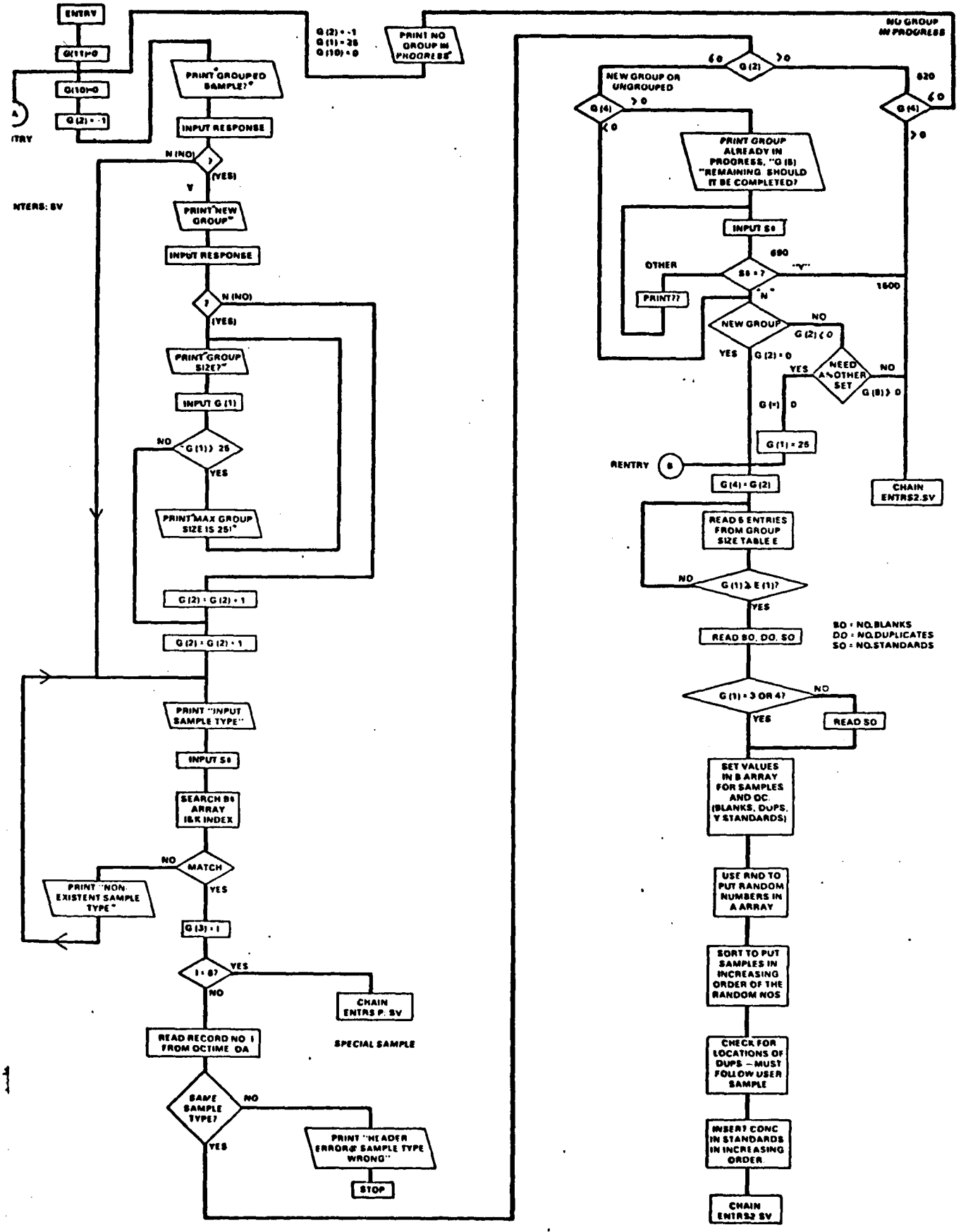
APPENDIX C

LOGIC FLOWCHARTS FOR MAJOR PROGRAMS

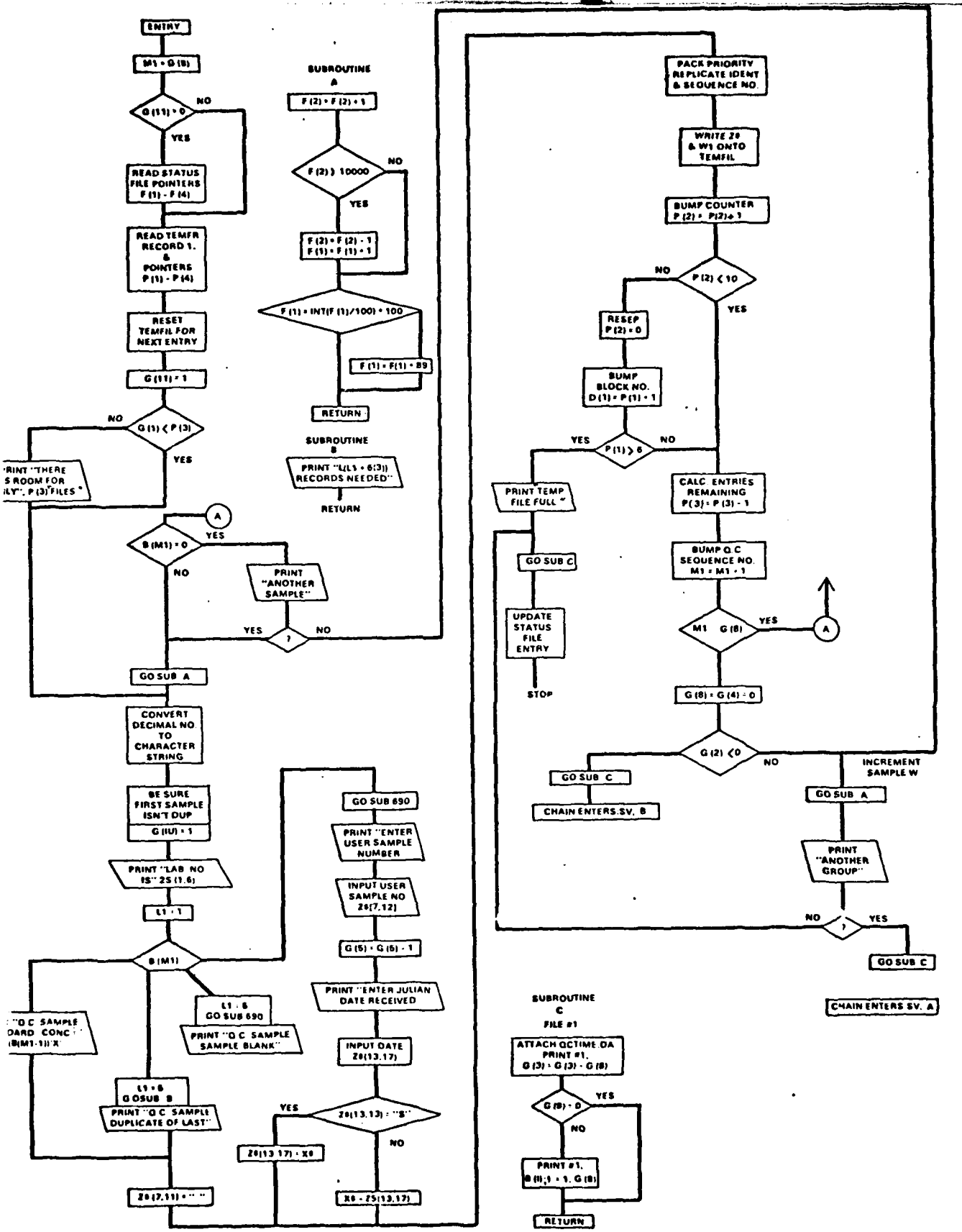
This section includes logic flowcharts for the two most complicated portions of the system. Flowcharts are included for these, because the logic was complicated enough that flowcharts were required for program development, and will probably be useful for further debugging and/or modification in laboratory use. The other programs were much simpler, or simply developed from portions of these programs, that flowcharts were not needed for their development.

SAMPLE LOGIN PROGRAMS

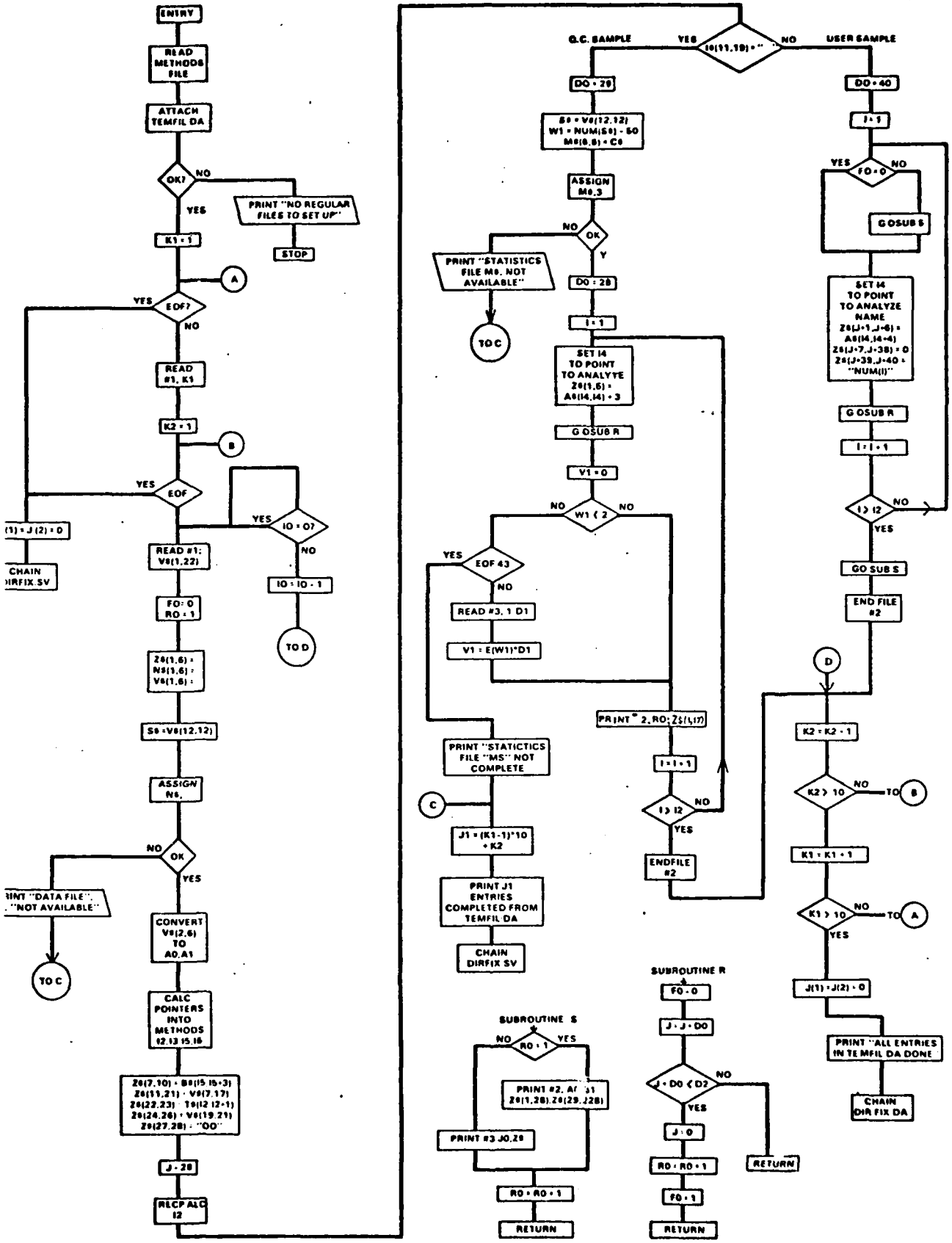
Three programs are in this set, ENTERS, ENTRS2, and FILWRT.



LOGIC FLOWCHART FOR ENTERS.SV



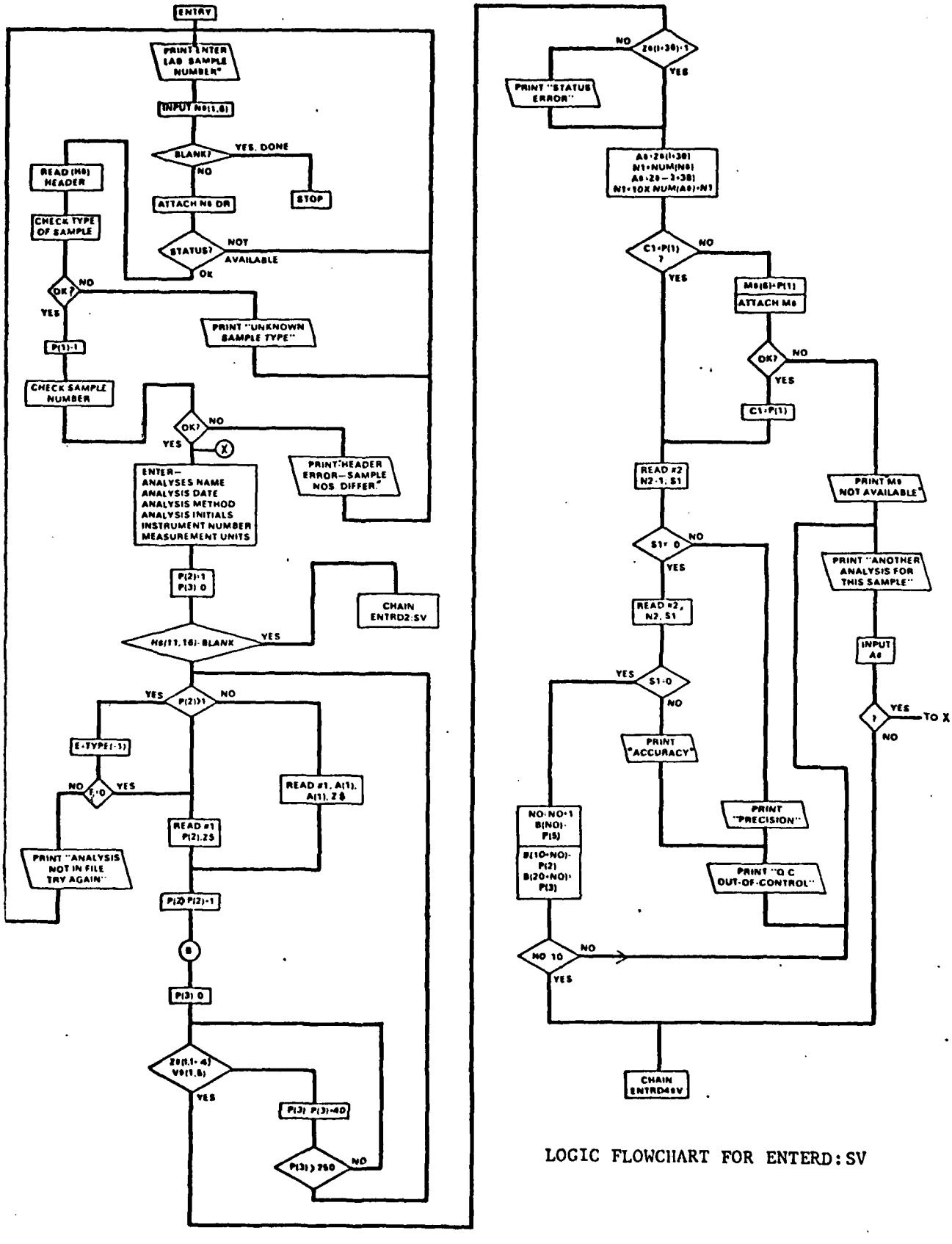
LOGIC FLOWCHART FOR ENTRS2:SV



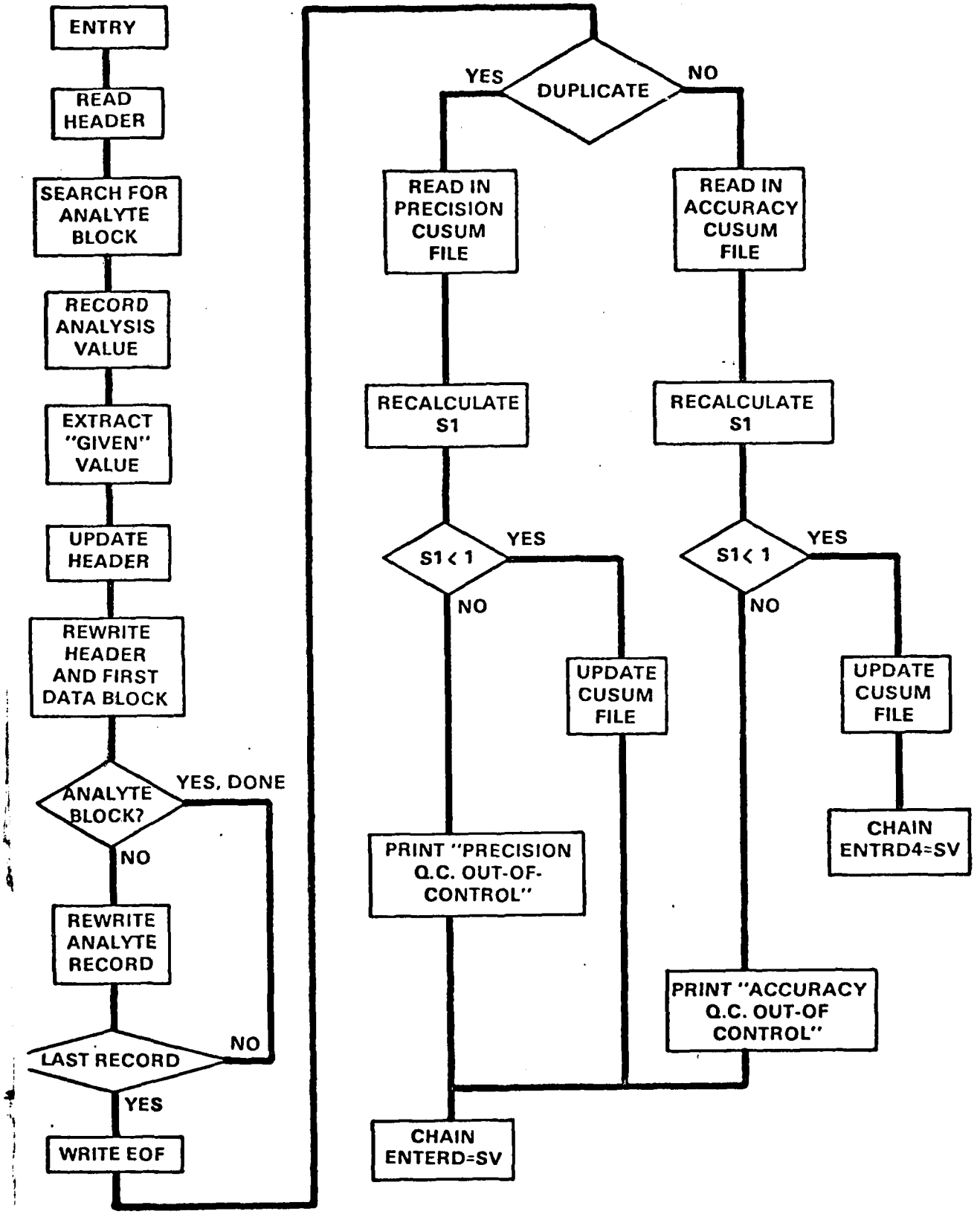
LOGIC FLOWCHART FOR FILWRT:SV

DATA ENTRY PROGRAMS

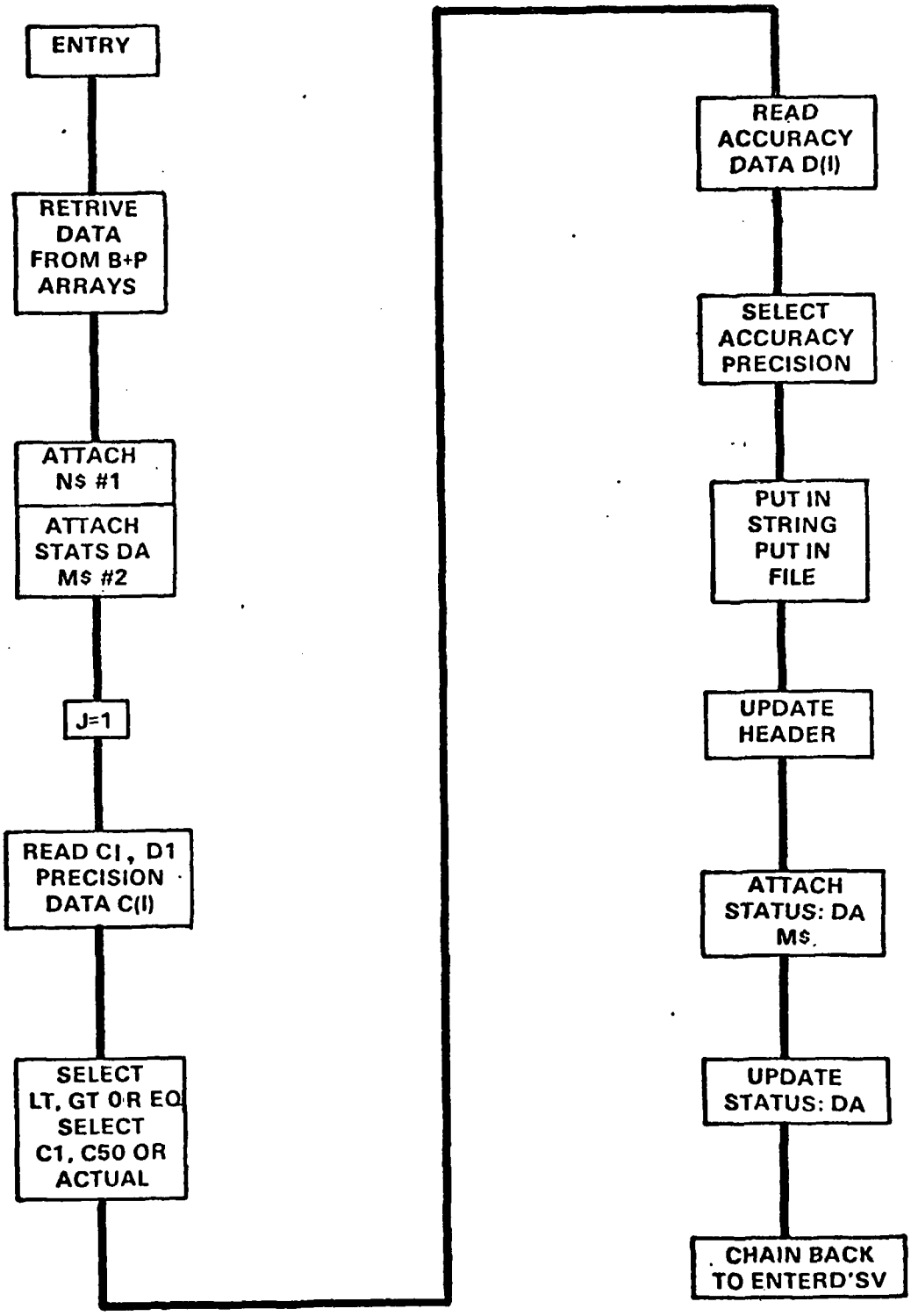
There are three programs in this set: ENTERD, ENTRD2, and ENTRD4.



LOGIC FLOWCHART FOR ENTERD:SV



LOGIC FLOWCHART FOR ENTRD2:SV



LOGIC FLOWCHART FOR ENTRD4


DAT
FILM
7-8