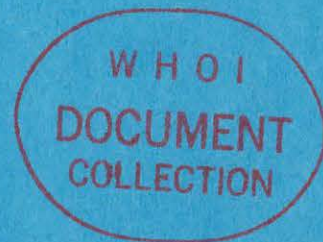


WHOI-81-88

copy 2

Woods Hole

*Oceanographic
Institution*



A COMPUTER PROGRAM TO PROCESS DATA FROM A
DIRECT CURRENT PLASMA EMISSION SPECTROMETER

by

Donald C. Bankston

October 1981

TECHNICAL REPORT

*Prepared for the U.S. Department of Energy,
under Contract DE-A102-81EV10694; and the
National Science Foundation under Grant OCE
81-19056.*

WOODS HOLE, MASSACHUSETTS 02543

WHOI-81-88

A COMPUTER PROGRAM TO PROCESS DATA FROM A
DIRECT CURRENT PLASMA EMISSION SPECTROMETER

by

Donald C. Bankston

WHOI
DOCUMENT
COLLECTION



WOODS HOLE OCEANOGRAPHIC INSTITUTION
Woods Hole, Massachusetts 02543

October 1981

TECHNICAL REPORT

*Prepared for the United States Department of Energy under
Contract DE-A102-81EV10694 and for the National Science
Foundation under Grant OCE 81-19056.*

*Reproduction in whole or in part is permitted for any pur-
pose of the United States Government. This report should
be cited as: Woods Hole Oceanog. Inst. Tech. Rept. WHOI-81-88.*

Approved for Distribution:

Geoffrey Thompson
Geoffrey Thompson, Chairman
Department of Chemistry

Abstract

This program processes data from a Spectraspan IIB spectrometer retrofitted with a Spectrajet III plasma excitation source (Spectrametrics, Inc., Andover, MA). The spectrometer is operated manually and can determine only one element at a time. Accordingly, the program treats data for each element individually.

Two of the spectrometer's performance characteristics tend to change gradually during normal operation: average blank count and analytical sensitivity. The program compensates for these variations. For the program to process data from the spectrometer, a reagent blank and a high standard (in this order) should be aspirated first; then the high standard and blank (in this order) should be run again after every six or fewer samples.

The program, moreover, causes all data to be converted to logarithmic form before being used in computation. This makes possible the calculation of statistically valid confidence limits about predicted analyte concentrations.

The program consists of a main section, DCPEQES, and three external sub-routines: BASELINE, SENSITIV, and XLNSQFT. BASELINE compensates for any change in the average blank count, and SENSITIV corrects for any variation in analytical sensitivity. XLNSQFT governs all correlation and prediction functions and has two interchangeable versions: one for first-order; the other for second-order, correlation.

Table of Contents

Documentation for Program DCPEOES 1

Source Listing of DCPEOES 3

Documentation for Subroutine BASELINE 4

Source Listing of BASELINE. 7

Documentation for Subroutine SENSITIV10

Source Listing of SENSITIV13

Documentation for Subroutine XLNSQFT16

Source Listing of XLNSQFT (First Order Calibration)18

Source Listing of XLNSQFT (Second Order Calibration).20

NAME: DCPEOES

TYPE: Main Program

PURPOSE: To derive concentrations of an element from counts generated by a dc plasma echelle emission spectrometer.

SPECTROMETER: Spectraspan IIB retrofitted with a Spectrajet III plasma source.

COMPUTER: Honeywell Sigma 7

SOURCE LANGUAGE: Fortran IV

PROGRAM CATEGORY: Statistics in Instrumental Analysis

DESCRIPTION: This program calls three subroutines that perform the following tasks respectively:

- 1) Estimation of an average blank count for every aspiration of a non-blank and deduction of this estimated blank count from each replicate count made for the non-blank. If the number of non-blanks between any two blanks exceeds 8, an appropriate error message is printed, and the program is terminated normally.
- 2) Further adjustment of the blank-corrected counts for any time-dependent variation in analytical sensitivity. If the number of samples or calibration standards between any two quality control standards exceeds 6, an appropriate error message is printed, and the program terminates normally.
- 3) Employment of the method of least squares, with a log/log data conversion, to define a first or second order calibration function to predict concentrations of the analyte in samples. For each set of replicate concentrations, the arithmetic mean and standard deviation are calculated.

INPUT: See descriptions of input to the different subroutines.

OUTPUT: See descriptions of output by the different subroutines.

USAGE: On-line

RESTRICTIONS: These are itemized for each subroutine.

LOAD MODULE STORAGE REQUIREMENTS: 10 K-words (40 k-bytes)

SUBPROGRAMS REQUIRED: BASELINE, SENSITIV, XLNSQFT

OPERATIONAL ENVIRONMENT: There is no external data input to the main program itself. The data control block for output is F:108 (terminal printer). To facilitate the portability of the main program, and of each subroutine, the code number(s) of any DCB(s) used is (are) specified only once and appear(s) in a DATA statement near the beginning of the file.

PROGRAM LOGIC: See Figure 1

TIMING: Not tested

ERRORS AND DIAGNOSTICS: These are itemized for each subroutine.

PROGRAMMER: Donald C. Bankston

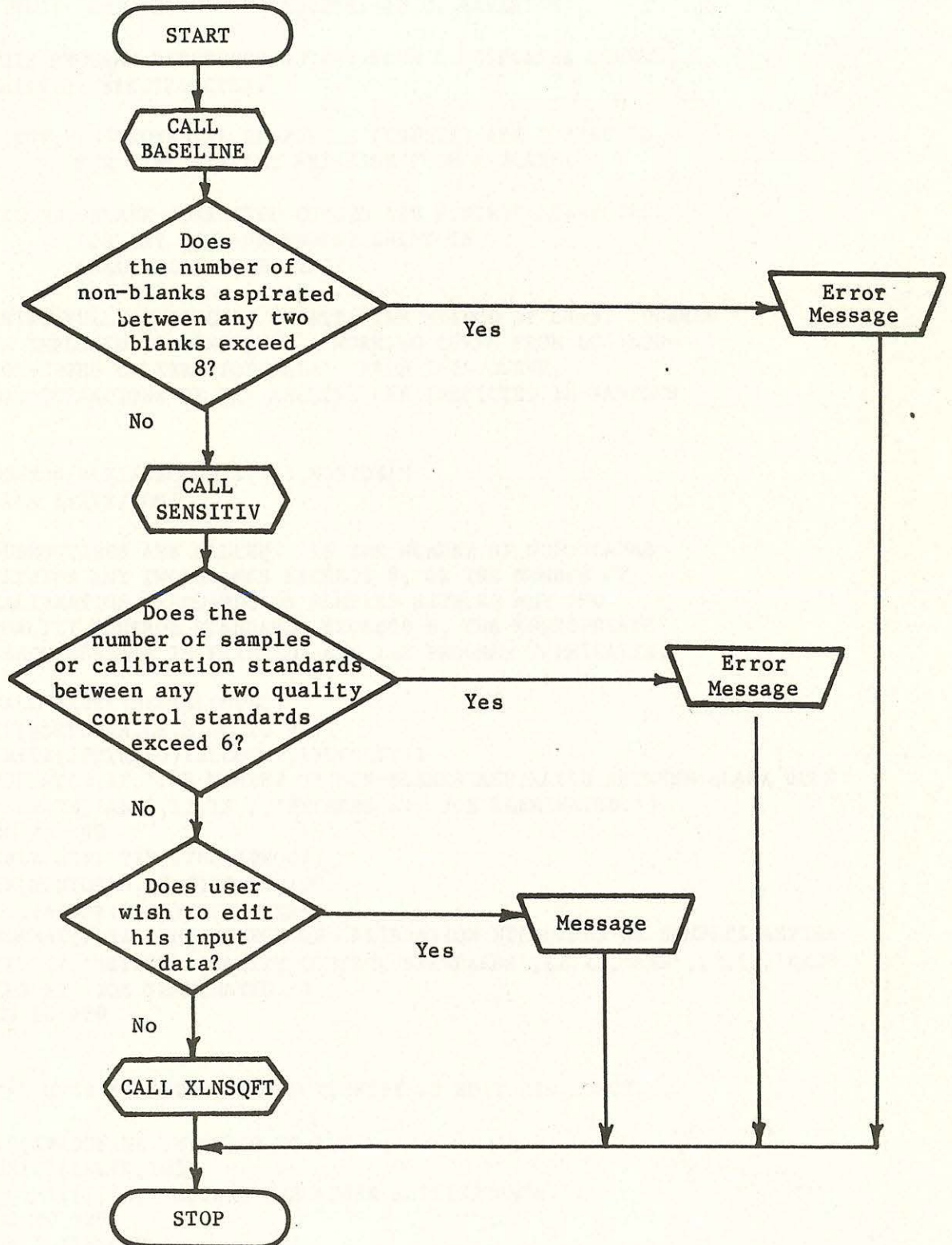
ORIGINATORS: Dr. Michael J. Mottl and Dr. Vaughan T. Bowen

DATE: June 5, 1981

Reference

Bankston, D. C., "Processing data from the dc plasma echelle emission spectrometer". American Laboratory, Vol. 13, p. 31, March 1981.

Figure 1: Flow Chart for DCPEOES



Source Listing of DCPEOES

C TITLE: DCPEOES. AUTHOR: D.C. BANKSTON
C
C THIS PROGRAM PROCESSES OUTPUT FROM A DC PLASMA ECHELLE
C EMISSION SPECTROMETER.
C
C FIRST, INSTRUMENTAL RESPONSES (COUNTS) ARE CORRECTED
C FOR TIME-VARIANT EMISSION FROM A BLANK.
C
C SECOND, BLANK-CORRECTED COUNTS ARE FURTHER CORRECTED
C FOR ANY TIME-DEPENDENT SHIFT IN
C ANALYTICAL SENSITIVITY.
C
C USING FULLY CORRECTED COUNTS, THE METHOD OF LEAST SQUARES
C IS EMPLOYED TO CONSTRUCT A WORKING CURVE FROM LOG/LOG-
C CONVERTED CALIBRATION DATA. FROM THIS CURVE,
C CONCENTRATIONS OF THE ANALYTE ARE PREDICTED IN SAMPLES.
C
C
C COMMON/BLK1/NBCOUNTS(74),NOSTDSAM
C DATA IPRTR/108/
C
C SUBROUTINES ARE CALLED. IF THE NUMBER OF NON-BLANKS
C BETWEEN ANY TWO BLANKS EXCEEDS 8, OR THE NUMBER OF
C CALIBRATION STANDARDS OR SAMPLES BETWEEN ANY TWO
C QUALITY CONTROL STANDARDS EXCEEDS 6, THE APPROPRIATE
C ERROR MESSAGE IS PRINTED, AND THE PROGRAM TERMINATES.
C
C CALL BASELINE(IBLCOUNT)
C IF(NOSTDSAM.LE.8)GO TO 90
C WRITE(IPRTR,60)IBLCOUNT,IBLCOUNT+1
60 FORMAT(/,1X,'THE NUMBER OF NON-BLANKS ASPIRATED BETWEEN BLANK RUNS
C ',I2,1X,'AND',I2,1X,/, 'EXCEEDS 8. JOB TERMINATED.')

GO TO 999
90 CALL SENSITIV(ITER,IPOOF)
C IF(NOSTDSAM.LE.6)GO TO 100
C WRITE(IPRTR,95)ITER,ITER+1
95 FORMAT(/,1X,'THE NUMBER OF CALIBRATION STANDARDS OR SAMPLES ASPIRA
C CTED',/, 'BETWEEN QUALITY CONTROL STANDARDS',I2,1X,'AND',I2,1X,'EXCE
C CEDS 6. JOB TERMINATED.')

GO TO 999
C
C THE USER IS GIVEN AN OPPORTUNITY TO EDIT HIS INPUT.
C
100 IF(IPOOF.NE.3HYES)GO TO 110
C WRITE(IPRTR,105)
105 FORMAT(/,1X,'RESTART JOB AFTER EDITING DATA.')

GO TO 999
110 CALL XLNSQFT
999 STOP
END

NAME: BASELINE

TYPE: Subroutine

PURPOSE: To estimate an average blank count for every aspiration of a non-blank and deduct this estimated blank count from each replicate count made for the non-blank.

COMPUTER: Honeywell Sigma 7

SOURCE LANGUAGE: FORTRAN IV

PROGRAM CATEGORY: Statistics in Instrumental Analysis

DESCRIPTION: In any succession of solutions aspirated into the nebulizer of the spectrometer, a reagent blank occurs at various points but always at the very beginning and at the very end. If more than 8 non-blanks are reported to intervene between any two consecutive blanks, then control is immediately returned to the main program; an appropriate error message is printed, and execution of the program is terminated normally.

Data for each pair of consecutive blanks, together with information on the sequence of non-blanks aspirated between them, is treated as an integral portion of the data input. It is assumed that any baseline shift occurring between successive blanks is linear. $(\bar{X}_{est})_n$, the estimated average blank count to be subtracted from every count reported for the Nth non-blank occurring between the two consecutive blanks, is given by

$$(\bar{X}_{est})_n = \bar{X}_1 - n \frac{\bar{X}_1 - \bar{X}_2}{N + 1}$$

where \bar{X}_1 denotes the mean count for the first blank,
 \bar{X}_2 indicates the mean count for the second blank, and
N denotes the number of non blanks between the two blanks.
 C_{nj} , the jth baseline-corrected count on the nth non-blank, is then given by

$C_{nj} = K_{nj} - (\bar{X}_{est})_n$
where K_{nj} denotes the jth raw count for the nth non-blank.

INPUT: (listed in its record-wise order of introduction):

- 1) The total number of aspirations of the blank. If the value entered exceeds 8, then the number of aspirations defaults to 8, an error message is printed, and execution of the program continues normally.

- 2) For each blank:
 - a) The number of counts per aspiration. If the value entered exceeds 10, then the number of counts defaults to 10, an error message is printed, and execution of the program continues normally.
 - b) The counts reported for the current aspiration.
- 3) The total number of non-blanks. If the value entered exceeds 8, then control is returned to the main program, and program execution is terminated normally.
- 4) For each non-blank:
 - a) The number of counts per aspiration. If the value entered exceeds 10, then the number of counts defaults to 10, an error message is printed, and execution of the program continues normally.
 - b) An alphanumeric identifier (cols. 1-4), an "analyte concentration", and the set of counts reported for the current non-blank. If the non-blank is a calibration standard, then the "analyte concentration" entered is the nominal one; but, if the non-blank is an unknown sample, then the value entered for "analyte concentration" is zero.
The alphanumeric non-blank identifier is entered in A4 format. All numeric data are introduced in widthless G-format.

OUTPUT: Besides questions or statements printed to the user, aspiration identifiers, nominal analyte concentrations, and background-corrected counts (as double-precision values) are stored in labeled common. Moreover, all input can be written onto a disk file.

USAGE: On-line.

RESTRICTIONS: These consist only of the dimensioning limitations detailed above in the INPUT section.

STORAGE REQUIREMENTS: 2,124 word locations.

SUBPROGRAMS REQUIRED: None.

OPERATIONAL ENVIRONMENT:

This program can be run in either of two modes: manual or automatic. In the manual mode, the user enters data, and this is then written onto a disk file that can be edited. In the automatic mode, edited data are read from the disk file and written out to the terminal printer. The allocation of data control blocks (DCBs) is as follows:

DCB	Manual Mode		Automatic Mode	
	READs from	WRITEs to	READs from	WRITEs to
F:105	Terminal Keyboard	----	Disk File	----
F:106	----	Disk File	----	Terminal Printer
F:108	----	Terminal Printer	----	Terminal Printer

TIMING: Not tested.

ERRORS AND DIAGNOSTICS: Self-explanatory.

PROGRAMMER: Donald C. Bankston

ORIGINATORS: Dr. Michael J. Mottl and Dr. Vaughan T. Bowen

DATE: June 5, 1981

PROGRAM LOGIC: See Figure 2

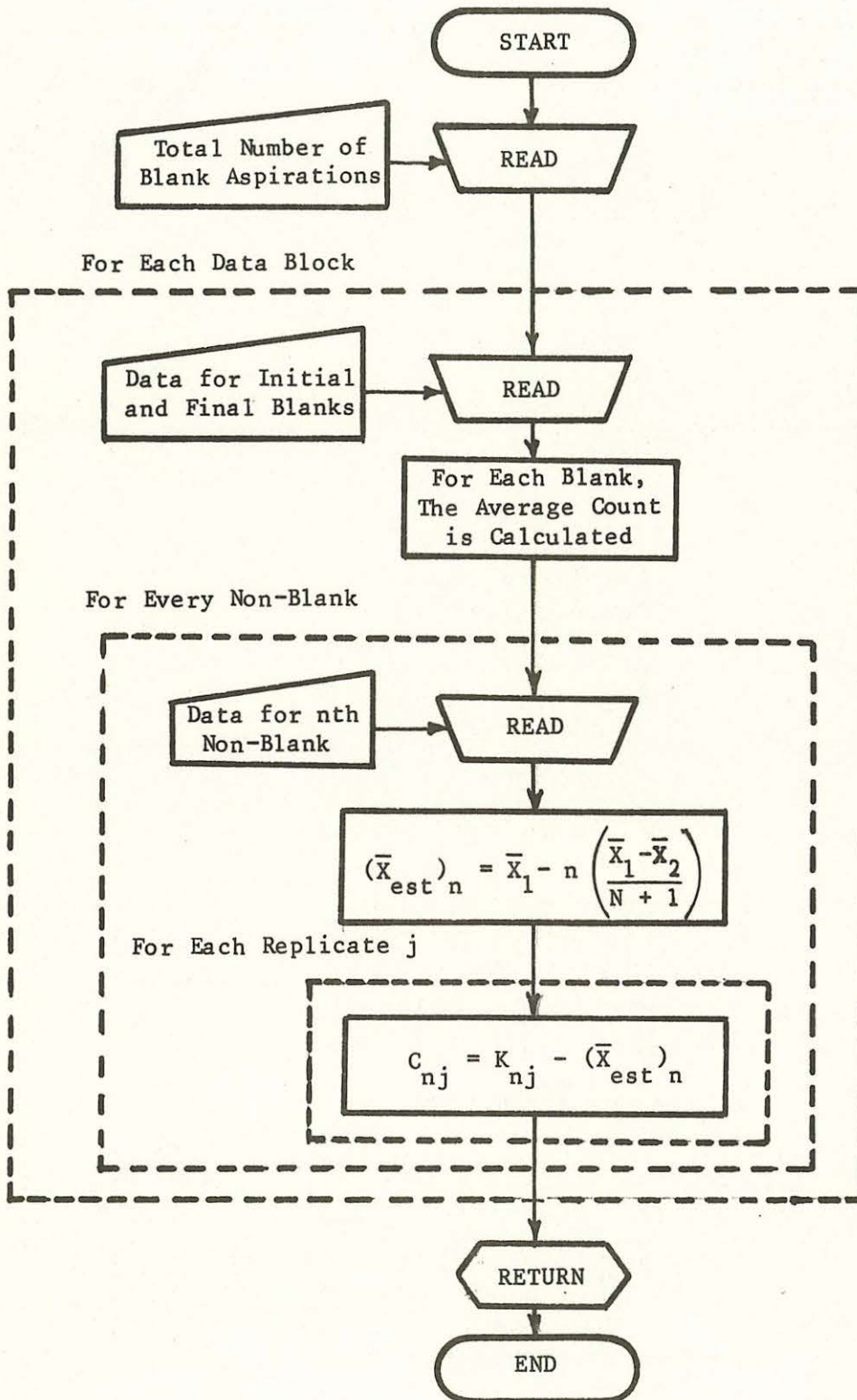


Figure 2: Flow Chart for BASELINE

SOURCE LISTING OF SUBROUTINE BASELINE

SUBROUTINE BASELINE(IBLCOUNT)

C
C
C
C
C
C
C
C
C

FOR EVERY ASPIRATION OF A QUALITY CONTROL STANDARD.
CALIBRATION STANDARD, OR SAMPLE, THIS SUBROUTINE
CALCULATES A BLANK COUNT AND DEDUCTS IT FROM EACH
NON-BLANK COUNT ASSOCIATED WITH THAT ASPIRATION.

VARIABLES ARE GROUPED AND TYPED, ARRAYS ARE
DIMENSIONED, AND DCB'S ARE NAMED.

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLK1/NBCOUNTS(74),NOSTDSAM
COMMON/BLK2/BCSTDSAM(74,10),IDSUP(74),SUPCONC(74),NPRESS
DIMENSION BLANK(10),STDSAM(74,10)
DATA INPUT/105/IPRTR/108/IDF/106/

C
C
C
C

NOBLKASP DENOTES THE TOTAL NUMBER OF ASPIRATIONS
OF THE BLANK.

WRITE(IPRTR,10)
10 FORMAT(1X,'WHAT IS THE TOTAL NUMBER OF ASPIRATIONS OF THE BLANK?')
READ(INPUT,15)NOBLKASP
15 FORMAT(G)
WRITE(IDF,16)NOBLKASP
16 FORMAT(I3)
IF(NOBLKASP.LE.8)GO TO 18
NOBLKASP=8
WRITE(IPRTR,17)
17 FORMAT(1X,'WARNING! THE TOTAL NUMBER OF ASPIRATIONS OF THE BLANK
CIS',/, 'LIMITED TO, AND HAS DEFAULTED TO, 8.')

C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C

INSTRUMENTAL RESPONSES (COUNTS) ARE PROCESSED IN
BLOCKS. EACH OF THESE IS BRACKETED BY RESPONSES
FROM A PAIR OF BLANKS.

NPRESS DENOTES THE TOTAL NUMBER OF QUALITY CONTROL
STANDARDS, CALIBRATION STANDARDS,
AND/OR SAMPLES FOR WHICH DATA HAD
BEEN ENTERED BEFORE PROCESSING OF
THE CURRENT DATA BLOCK WAS BEGUN.

AVBLKI DENOTES THE MEAN COUNT FOR THE BLANK
IMMEDIATELY PRECEDING THE CURRENT
DATA BLOCK (THE IBLCOUNT-TH).

NOBLMEAS DENOTES THE NUMBER OF COUNTS ENTERED
FOR THE CURRENT BLANK.

18 NPRESS=0

```
DO 85 IBLCOUNT=1,NOBLKASP-1
  IF(IBLCOUNT.NE.1)GO TO 40
  WRITE(IPRTR,20)IBLCOUNT
20 FORMAT(/,1X,'HOW MANY MEASUREMENTS ON ASPIRATION NO.',I2,1X,'OF TH
  CE BLANK?')
  READ(INPUT,15)NOBLMEAS
  WRITE(IDF,16)NOBLMEAS
  IF(NOBLMEAS.LE.10)GO TO 24
  NOBLMEAS=10
  WRITE(IPRTR,22)IBLCOUNT
22 FORMAT(/,1X,'WARNING! THE NUMBER OF MEASUREMENTS ON ASPIRATION NO
  C.',1X,I2,/, 'OF THE BLANK HAS BEEN LIMITED TO 10.')
```

```
24 WRITE(IPRTR,25)IBLCOUNT
25 FORMAT(/,1X,'WHAT ARE THE MEASUREMENTS ON ASPIRATION NO.',I2,1X,'O
  CF THE BLANK?')
  READ(INPUT,30)NOBLMEAS,(BLANK(NBLREPS),NBLREPS=1,NOBLMEAS)
30 FORMAT(NG)
  WRITE(IDF,31)NOBLMEAS,(BLANK(NBLREPS),NBLREPS=1,NOBLMEAS)
31 FORMAT(NF6.0)
  SUMBLANK=0.
  DO 35 NBLREPS=1,NOBLMEAS
  SUMBLANK=SUMBLANK+BLANK(NBLREPS)
35 CONTINUE
  AVBLKI=SUMBLANK/NOBLMEAS
  GO TO 45
40 AVBLKI=AVBLKF
      WHERE AVBLKF DENOTES THE MEAN COUNT FOR THE BLANK
      TERMINATING THE DATA BLOCK
      ENTERED JUST PREVIOUSLY.
C
C
C
C
45 WRITE(IPRTR,20)IBLCOUNT+1
  READ(INPUT,15)NOBLMEAS
  WRITE(IDF,16)NOBLMEAS
  IF(NOBLMEAS.LE.10)GO TO 47
  NOBLMEAS=10
  WRITE(IPRTR,22)IBLCOUNT+1
47 WRITE(IPRTR,25)IBLCOUNT+1
  READ(INPUT,30)NOBLMEAS,(BLANK(NBLREPS),NBLREPS=1,NOBLMEAS)
  WRITE(IDF,31)NOBLMEAS,(BLANK(NBLREPS),NBLREPS=1,NOBLMEAS)
  SUMBLANK=0
  DO 50 NBLREPS=1,NOBLMEAS
  SUMBLANK=SUMBLANK+BLANK(NBLREPS)
50 CONTINUE
  AVBLKF=SUMBLANK/NOBLMEAS
      WHERE AVBLKF DENOTES THE MEAN COUNT FOR THE BLANK
      TERMINATING THE CURRENT DATA
      BLOCK.
C
C
C
  WRITE(IPRTR,53)IBLCOUNT,IBLCOUNT+1
53 FORMAT(//,1X,'HOW MANY ASPIRATIONS LIE BETWEEN BLANK RUNS',I2,1X,
  C'AND',I2,'?')
  READ(INPUT,15)NOSTDSAM
```

```
WRITE(IDF,16)NOSTDSAM
WRITE(IPRTR,55)
55 FORMAT(//)
IF(NOSTDSAM.GT.8)GO TO 999
AVBKDRFT=(AVBLKI-AVBLKF)/(NDSTDSAM+1)

C
C      STDSAM DENOTES THE MATRIX OF UNCORRECTED COUNTS
C      OBTAINED FOR NON-BLANKS IN ALL
C      DATA BLOCKS COMBINED.
C
C      BCSTDSAM DENOTES THE MATRIX OF BACKGROUND-CORREC-
C      -TED COUNTS OBTAINED FOR NON-
C      BLANKS IN ALL DATA BLOCKS COMBINED.
C
C      NBCOUNTS DENOTES A COLUMN VECTOR OF NUMBERS OF
C      COUNTS PER ROW OF BCSTDSAM.
C
C      IDSUP DENOTES A COLUMN VECTOR OF QUALITY CONTROL
C      STANDARD, SAMPLE, AND CALIBRATION STANDARD
C      IDENTIFIERS CORRESPONDING TO THE ROWS OF BCSTDSAM.
C
C      SUPCONC DENOTES A COLUMN VECTOR OF CONCENTRATIONS
C      OF THE ANALYTE.
C

57 FORMAT(A,G,NG)
58 FORMAT(A4,F5.0,NF6.0)
60 FORMAT(1X,'HOW MANY COUNTS FOR NON-BLANK NO.',I2,'?')
63 FORMAT(1X,'WHAT ARE THE DATA FOR NON-BLANK NO.',I2,'?')
DO 75 I=1,NOSTDSAM
WRITE(IPRTR,60)NPRESS+I
READ(INPUT,15)NBCOUNTS(NPRESS+I)
WRITE(IDF,16)NBCOUNTS(NPRESS+I)
IF(NBCOUNTS(NPRESS+I).LE.10)GO TO 65
NBCOUNTS(NPRESS+I)=10
WRITE(IPRTR,60)NPRESS+I
64 FORMAT(1X,'WARNING! THE NUMBER OF MEASUREMENTS ON NON-BLANK NO.',
C1X,I3,/, 'HAS BEEN LIMITED TO 10.')
65 WRITE(IPRTR,63)NPRESS+I)
READ(INPUT,57)IDSUP(NPRESS+I),SUPCONC(NPRESS+I),NBCOUNTS(NPRESS+I)
C,(STDSAM(NPRESS+I,J),J=1,NBCOUNTS(NPRESS+I))
WRITE(IDF,58)IDSUP(NPRESS+I),SUPCONC(NPRESS+I),NBCOUNTS(NPRESS+I),
C(STDSAM(NPRESS+I,J),J=1,NBCOUNTS(NPRESS+I))
AVBLKI=AVBLKI-AVBKDRFT
DO 70 J=1,NBCOUNTS(NPRESS+I)
BCSTDSAM(NPRESS+I,J)=STDSAM(NPRESS+I,J)-AVBLKI
70 CONTINUE
75 CONTINUE
NPRESS=NPRESS+NOSTDSAM
85 CONTINUE
999 RETURN
END
```

NAME: SENSITIV

TYPE: Subroutine

PURPOSE: To adjust blank-corrected counts for any time-dependent variation in analytical sensitivity.

COMPUTER: Honeywell Sigma 7

SOURCE LANGUAGE: FORTRAN IV

PROGRAM CATEGORY: Statistics in instrumental analysis.

DESCRIPTION: If reagent blanks are disregarded, then, in any succession of solutions aspirated into the nebulizer of the spectrometer, a high standard used as a control monitor occurs at various points, but always at the very beginning and at the very end. If more than 6 samples or calibration standards are reported to intervene between any two consecutive aspirations of the monitor, then control is immediately returned to the main program, an appropriate error message is printed, and execution of the program is terminated normally.

Data for each pair of consecutive control monitors, and for the sequence of samples and standards aspirated between them, is treated as a single unit of information. It is assumed that any variation in sensitivity occurring between successive monitors is linear. $(\bar{Z}_{est})_n$, the estimated average monitor count associated with the set of counts reported for the nth sample or standard occurring between the two consecutive monitors, is given by

$$(\bar{Z}_{est})_n = \bar{Z}_1 - n \frac{\bar{Z}_1 - \bar{Z}_2}{N + 1}$$

where \bar{Z}_1 denotes the mean count for the first control monitor, \bar{Z}_2 denotes the mean count for the second control monitor, and N denotes the number of standards and samples between the two monitors.

S_{nj} , the j-th sensitivity-corrected count for the n-th standard or sample in the current data block, is then given by

$$S_{nj} = C_{nj} \frac{\bar{Z}_0}{(\bar{Z}_{est})_n}$$

in which C_{nj} denotes the j-th baseline-corrected count on the n-th sample or standard, and \bar{z}_0 denotes the average count for the first control monitor in the first data block.

INPUT:

- 1) The total number of aspirations of the control monitor. If the value entered exceeds 8, then the number of aspirations defaults to 8, an error message is printed, and execution of the program continues normally.
- 2) The number of standard or samples aspirated between each pair of consecutive control monitors. If any of these exceeds 6, control returns to the main program, an appropriate error message is printed, and execution of the program terminates normally.
- 3) The user is prompted by the question:

"DO YOU WISH TO PROOFREAD YOUR DATA INPUT?"

If the answer to this is affirmative, the word YES is entered, control returns to the main program, and execution of the program terminates normally. If the answer is negative, anything other than YES may be typed in, and program execution continues.

- 4) Aspiration identifiers, nominal analyte concentrations, and background corrected counts read out of BASELINE and stored in labeled COMMON.

Items 1 and 2 are entered as integers, in widthless G-format. Item 3 is alphanumeric data entered in widthless A-format, and the background corrected counts mentioned in item 4 are double precision.

OUTPUT: Besides questions and statements for the user, two tables of data are produced. Both of these contain identifiers and nominal analyte concentrations, but the first table contains baseline (Y-intercept)-corrected counts, while the second table contains sensitivity (slope)-corrected counts. In each table, the identifiers are in A5-format, the concentrations in F8.2, and the counts in F6.0.

USAGE: On-line.

RESTRICTIONS: These consist only of the dimensioning limitations detailed above in the INPUT section.

STORAGE REQUIREMENTS: 483 word locations.

SUBPROGRAMS REQUIRED: None.

OPERATIONAL ENVIRONMENT: Same as for BASELINE.

TIMING: Not tested.

ERRORS AND DIAGNOSTICS: Self-explanatory.

PROGRAMMER: Donald C. Bankston

ORIGINATORS: Dr. Michael J. Mottl and Dr. Vaughan T. Bowen

DATE: June 5, 1981

PROGRAM LOGIC: See Figure 3.

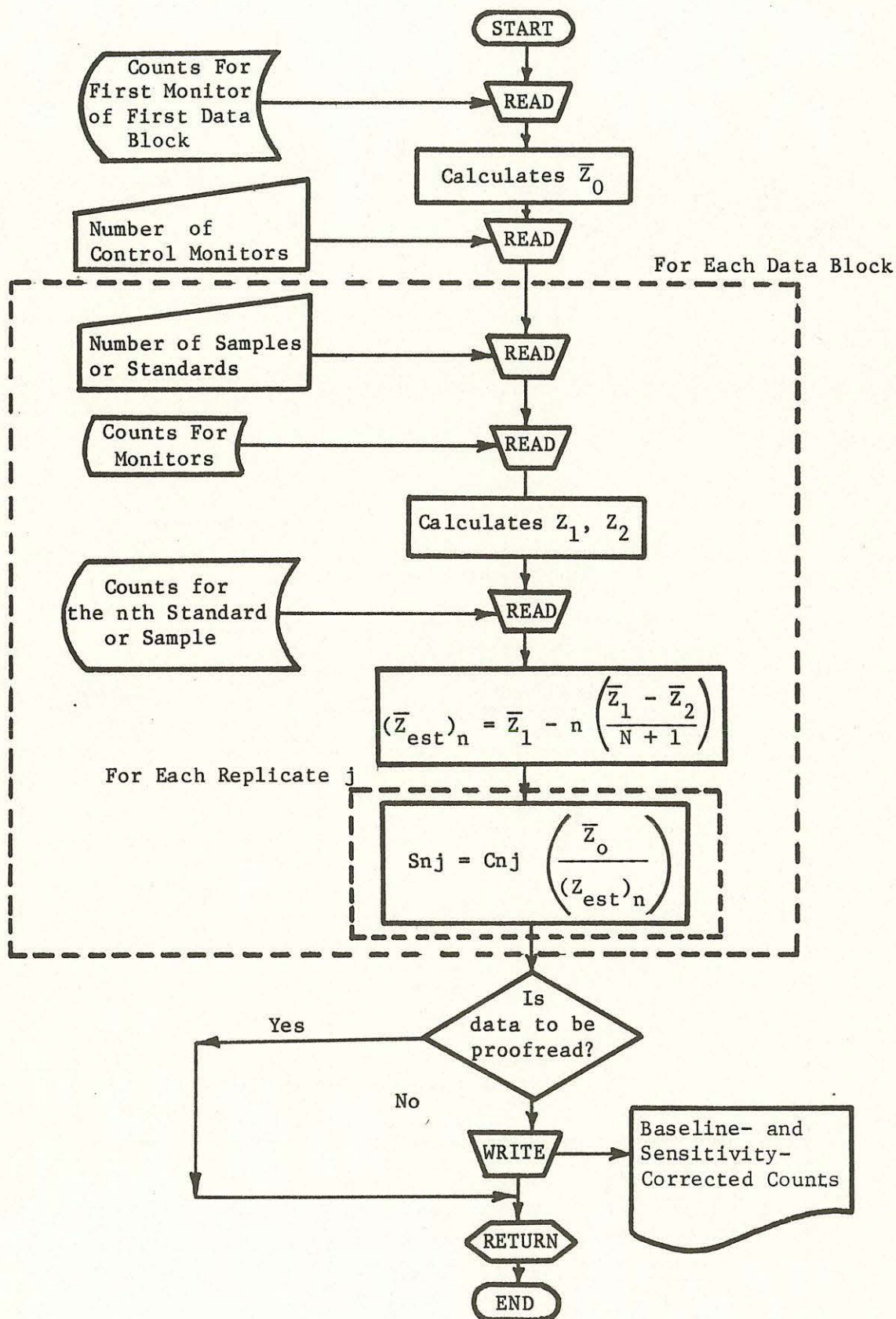


Figure 3: Flow Chart For SENSITIV

SOURCE LISTING FOR SUBROUTINE SENSITIV

```

SUBROUTINE SENSITIV(ITER,IPROOF)
C
C   FOR EACH ASPIRATION OF A SAMPLE OR CALIBRATION
C   STANDARD, THIS SUBROUTINE CALCULATES A QUALITY CONTROL
C   STANDARD COUNT AND MULTIPLIES EACH SAMPLE OR CALIBRATION
C   STANDARD COUNT BY THE RATIO OF AN INITIAL QUALITY
C   CONTROL STANDARD COUNT TO THE CALCULATED QUALITY
C   CONTROL STANDARD COUNT.
C
C   VARIABLES ARE GROUPED AND TYPED, ARRAYS ARE
C   DIMENSIONED, AND DCB'S ARE NAMED.
C
C   IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLK1/NBCOUNTS(74),NOSTDSAM
COMMON/BLK2/BCSTDSAM(74,10),IDSUP(74),SUPCONC(74),NPRESS
COMMON/BLK3/DCSTDSAM(60,10),NDCOUNTS(60),SUBCONC(60),IDSUB(60),IDC
CROW
DATA INPUT/105/IPRTR/108/IDF/106/
C
C   DCSTDSAM DENOTES THE MATRIX OF COUNTS CORRECTED FIRST
C   FOR BACKGROUND, THEN FOR SHIFT IN
C   ANALYTICAL SENSITIVITY.
C
C   XBARO DENOTES THE AVERAGE COUNT FOR THE REFERENCE
C   QUALITY CONTROL STANDARD.
C
C   SUMMON=0.
DO 10 MONREPS=1,NBCOUNTS(1)
SUMMON=SUMMON+BCSTDSAM(1,MONREPS)
10 CONTINUE
XBARO=SUMMON/NBCOUNTS(1)
C
C   AS IN BASELINE, COUNTS ARE PROCESSED IN BLOCKS.
C   THIS TIME, EACH BLOCK IS BRACKETED BY RESPONSES FROM
C   A PAIR OF CONSECUTIVELY RUN QUALITY CONTROL STANDARDS.
C   IT IS ASSUMED THAT DATA FOR THE REFERENCE QUALITY
C   CONTROL STANDARD COMPRISES THE FIRST ROW OF
C   BCSTDSAM.
C
C   THE IDCROW-TH ROW OF DCSTDSAM CORRESPONDS TO THE
C   IBCROW-TH ROW OF BCSTDSAM.
C
C   NOMNTR DENOTES THE TOTAL NUMBER OF ASPIRATIONS OF
C   THE QUALITY CONTROL STANDARD.
C
C   MONITOR1=1
IDCROW=0
WRITE(IPRTR,15)
```

```
15 FORMAT(///,1X,'HOW MANY TIMES WAS THE QUALITY CONTROL STANDARD ASP
    CIRATED?')
    READ(INPUT,20)NOMNTR
20 FORMAT(G)
    WRITE(IDF,22)NOMNTR
22 FORMAT(I3)
    IF(NOMNTR.LE.8)GO TO 24
    NOMNTR=8
    WRITE(IPRTR,23)
23 FORMAT(1X,'WARNING! THE NUMBER OF ASPIRATIONS OF THE QUALITY CONT
    CROL STANDARD IS LIMITED TO, AND HAS DEFAULTED TO, 8.')
```

```
24 DO 55 ITER=1,NOMNTR-1
    WRITE(IPRTR,25)ITER,ITER+1
```

```
25 FORMAT(1X,'HOW MANY CALIBRATION STANDARDS OR SAMPLES WERE ASPIRATE
    CD BETWEEN QUALITY CONTROL STANDARDS',I2,1X,'AND',I2,'?')
```

C
C
C
C
C
C
C
C
C
C
C
C
C
C

XBAR1 DENOTES THE AVERAGE COUNT FOR THE QUALITY CONTROL STANDARD INITIATING THE CURRENT DATA BLOCK.

NOSTDSAM DENOTES THE NUMBER OF CALIBRATION STANDARDS OR SAMPLES IN THE CURRENT DATA BLOCK.

DATA FOR THE QUALITY CONTROL STANDARD TERMINATING THE CURRENT DATA BLOCK COMPRISE THE MONITOR2-TH ROW OF BCSTDSAM.

```
    READ(INPUT,20)NOSTDSAM
    WRITE(IDF,22)NOSTDSAM
    IF(NOSTDSAM.GT.6)GO TO 999
    MONITOR2=MONITOR1+NOSTDSAM+1
    IF(ITER.GT.1)GO TO 30
    XBAR1=XBARO
    GO TO 35
```

```
30 XBAR1=XBAR2
```

C
C
C
C
C

WHERE XBAR2 DENOTES THE AVERAGE COUNT FOR THE QUALITY CONTROL STANDARD TERMINATING THE IMMEDIATELY PREVIOUS DATA BLOCK.

```
35 SUMMON=0.
    DO 40 MONREPS=1,NBCOUNTS(MONITOR2)
    SUMMON=SUMMON+BCSTDSAM(MONITOR2,MONREPS)
```

```
40 CONTINUE
```

C
C
C
C

XBAR2=SUMMON/NBCOUNTS(MONITOR2)
WHERE XBAR2 DENOTES THE AVERAGE COUNT FOR THE QUALITY CONTROL STANDARD TERMINATING THE CURRENT DATA BLOCK.

```
AVSNDRFT=(XBAR1-XBAR2)/(NOSTDSAM+1)
IBCROW=MONITOR1
DO 50 INTER=1,NOSTDSAM
IDCROW=IDCROW+1
IBCROW=IBCROW+1
NDCOUNTS(IDCROW)=NBCOUNTS(IBCROW)
SUBCONC(IDCROW)=SUPCONC(IBCROW)
IDSUB(IDCROW)=IDSUP(IBCROW)
XBAR1=XBAR1-AVSNDRFT
DO 45 MONREPS=1,NBCOUNTS(IBCROW)
DCSTDSAM(IDCROW,MONREPS)=BCSTDSAM(IBCROW,MONREPS)+XBARO/XBAR1
45 CONTINUE
50 CONTINUE
MONITOR1=MONITOR2
55 CONTINUE
C
C   AT THIS POINT, PROGRAM EXECUTION MAY BE HALTED
C   TO ALLOW PROOFREADING OF THE DATA INPUT.
C
WRITE(IPRTR,56)
56 FORMAT(//,1X,'DO YOU WISH TO PROOFREAD YOUR DATA INPUT?')
READ(INPUT,57,END=119)IPROOF
57 FORMAT(A)
IF(IPROOF.EQ.3HYES)GO TO 999
C
C   AT THIS POINT, ALL DATA OUTPUT FROM
C   BASELINE AND SENSITIV ARE PRINTED.
C
119 WRITE(IPRTR,120)
120 FORMAT(//,2X,'IDENT',3X,'CONC',4X,'Y-INTERCEPT-CORRECTED COUNTS')
WRITE(IPRTR,125)
125 FORMAT(2X,'-----',3X,'----',4X,'- -----',/)
DO 130 I=1,NPRESS
WRITE(IPRTR,127)IDSUP(I),SUPCONC(I),NBCOUNTS(I),(BCSTDSAM(I,J),J=1
C,NBCOUNTS(I))
127 FORMAT(1X,A5,F8.2,NF6.0)
130 CONTINUE
WRITE(IPRTR,160)
160 FORMAT(//,2X,'IDENT',3X,'CONC',4X,'SLOPE-CORRECTED COUNTS')
WRITE(IPRTR,165)
165 FORMAT(2X,'-----',3X,'----',4X,'-----',/)
170 FORMAT(1X,A5,F8.2,NF6.0)
DO 175 I=1,IDCROW
WRITE(IPRTR,170)IDSUB(I),SUBCONC(I),NDCOUNTS(I),(DCSTDSAM(I,J),J=1
C,NDCOUNTS(I))
175 CONTINUE
999 RETURN
END
```

NAME: XLNSQFT

TYPE: Subroutine

PURPOSE: To derive a first or second order least squares calibration function from log/log transformed data, calculate concentrations of the analyte in samples, and, for each set of replicate determinations, calculate the mean and standard deviation.

COMPUTER: Honeywell Sigma 7

SOURCE LANGUAGE: FORTRAN IV

PROGRAM CATEGORY: Statistics in instrumental analysis.

PROGRAM DESCRIPTION: If the nominal concentration appearing in a row of non-blank data is non-zero, then this data row is understood to belong to a calibration standard; otherwise, to an analytical sample. This provision makes it possible to aspirate samples and standards in random order.

Instrument counts and nominal analyte concentrations for calibration standards undergo logarithmic transformation, and the method of least squares is used to calculate the coefficients of a first or second order calibration function from the transformed data.

This function is then used to estimate the concentration of the analyte in each sample. For each set of replicate determinations, the mean and standard deviation are calculated.

INPUT: The table of run identifiers, nominal concentrations, and sensitivity-corrected counts, together with its dimensioning parameters, produced by subroutine SENSITIV.

OUTPUT: 1) A table of calibration data in which standard identifiers are in A4 Format, nominal analyte concentrations in F7.2 Format, and counts in F5.0 Format.
2) A table of analytical results in which sample identifiers are in A5 Format, individual concentrations in F6.0 Format, mean concentrations in F7.2 Format, and standard deviations in F6.2 Format.

USAGE: On-line

RESTRICTIONS: None

STORAGE REQUIREMENTS: For first order calibration, 1903 word locations
For second order calibration, 2113 word locations.

SUBPROGRAMS REQUIRED: None

OPERATIONAL ENVIRONMENT: Same as for DCPEOES.

TIMING: Not tested

ERROR DIAGNOSTICS: None

PROGRAMMER: Don Bankston

ORIGINATORS: Dr. Michael J. Mottl and Dr. Vaughan T. Bowen

DATE: June 5, 1981

PROGRAM LOGIC: See Figure 4.

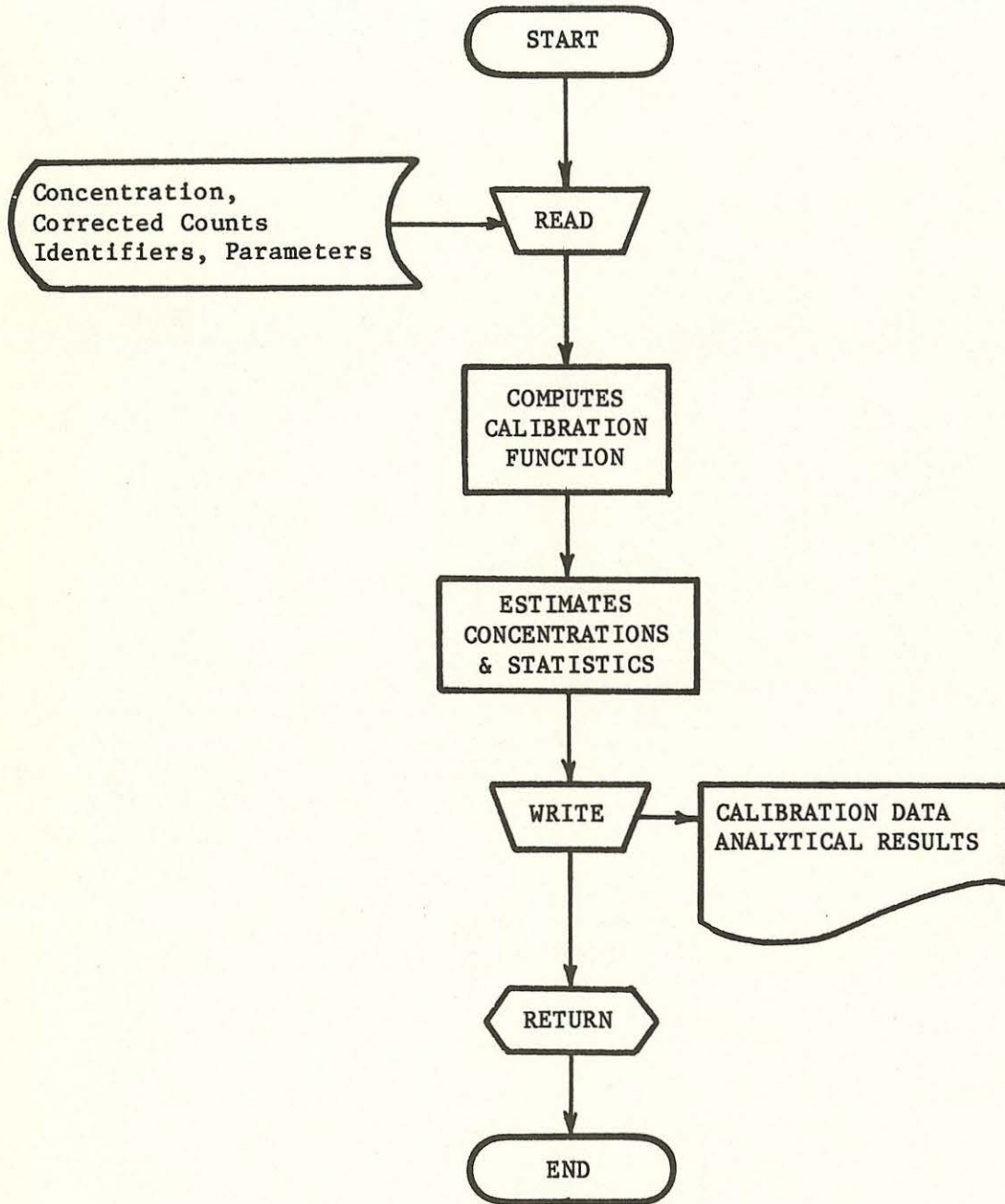


Figure 4: Flow Chart for XLNSQFT

SOURCE LISTING FOR SUBROUTINE XLNSQFT (First Order Calibration)

SUBROUTINE XLNSQFT

```
C
C   A LOG/LOG DATA TRANSFORMATION IS MADE, AND THE METHOD OF
C   LEAST SQUARES IS USED TO CALCULATE THE COEFFICIENTS OF A
C   LINEAR CALIBRATION CURVE FROM THE TRANSFORMED DATA.
C   THE UNTRANSFORMED CALIBRATION DATA SET, TOGETHER WITH
C   THE ANTILOGS OF PREDICTED Y-VALUES, ARE PRINTED OUT.
C   FINALLY, CONCENTRATIONS OF THE ANALYTE IN SAMPLES ARE
C   PRINTED.
C
C   VARIABLES ARE GROUPED AND TYPED, ARRAYS ARE
C   DIMENSIONED, AND DCB'S ARE NAMED.
C
C   IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLK3/DCSTDSAM(60,10),NDCOUNTS(60),SUBCONC(60),IDSUB(60),IDC
CROW
DIMENSION CALSTD(60,10),IDSTD(60),STDCONC(60),X(10),NSCOUNTS(60)
DATA INPUT/105/IPRTR/108/
C
C   CALIBRATION STANDARDS ARE SORTED OUT FROM SAMPLES, AND
C   SUMMATIONS ARE PERFORMED.
C
SUMY=0.
SUMXY=0.
SUMX=0.
SUMX2=0.
N=0
NROWS=0
DO 20 I=1, IDCROW
IF(SUBCONC(I).EQ.0.)GO TO 20
N=N+NDCOUNTS(I)
NROWS=NROWS+1
DO 10 J=1, NDCOUNTS(I)
SUMY=SUMY+DLOG(DCSTDSAM(I,J))
SUMXY=SUMXY+DLOG(SUBCONC(I))*DLOG(DCSTDSAM(I,J))
SUMX=SUMX+DLOG(SUBCONC(I))
SUMX2=SUMX2+(DLOG(SUBCONC(I)))**2
CALSTD(NROWS,J)=DCSTDSAM(I,J)
10 CONTINUE
IDSTD(NROWS)=IDSUB(I)
STDCONC(NROWS)=SUBCONC(I)
NSCOUNTS(NROWS)=NDCOUNTS(I)
20 CONTINUE
C
C   THE REGRESSION COEFFICIENTS ARE CALCULATED
C

$$A1 = (N * SUMXY - SUMX * SUMY) / (N * SUMX2 - SUMX ** 2)$$


$$A0 = (SUMY - A1 * SUMX) / N$$

C
```

```
C      CALIBRATION DATA ARE PRINTED OUT
C
      WRITE(IPRTR,22)
22  FORMAT(///,16X,'CALIBRATION DATA')
      WRITE(IPRTR,23)
23  FORMAT(16X,'----- ----',/)
      WRITE(IPRTR,25)
25  FORMAT(2X,'STD',3X,'STD',T50,'PREDICTED')
      WRITE(IPRTR,30)
30  FORMAT(1X,'IDENT',2X,'CONC',5X,'SLOPE-CORRECTED COUNTS',12X,'COUNT
      CS')
      WRITE(IPRTR,35)
35  FORMAT(1X,'-----',2X,'----',5X,'----- -----',12X,'-----
      C-',/)
40  FORMAT(2X,A4,1X,F7.2,3X,NF5.0,T50,F5.0)
      DO 45 I=1,NROWS
      WRITE(IPRTR,40)IDSTD(I),STDCONC(I),NSCOUNTS(I),(CALSTD(I,J),J=1,NS
      CCOUNTS(I)),DEXP(A0+A1*DLOG(STDCONC(I)))
45  CONTINUE
      WRITE(IPRTR,50)A0,A1
50  FORMAT(//,1X,'A0=',F10.7,2X,'A1=',F10.7)

C
C      SAMPLES ARE CULLED FROM CALIBRATION STANDARDS, AND
C      ANALYTE CONCENTRATIONS (WITH MEAN AND STANDARD
C      DEVIATION) ARE CALCULATED.
C
      WRITE(IPRTR,52)
52  FORMAT(///,16X,'ANALYTICAL RESULTS')
      WRITE(IPRTR,53)
53  FORMAT(1X,'----- -----',/)
      WRITE(IPRTR,55)
55  FORMAT(2X,'IDENT',4X,'ANALTYE CONCENTRATIONS')
      WRITE(IPRTR,60)
60  FORMAT(2X,'-----',4X,'----- -----')
65  FORMAT(1X,A5,NF6.0,2X,'MEAN=',F7.2,2X,'SDEV=',F6.2)
      DO 75 I=1, IDCROW
      IF(SUBCONC(I).NE.0.)GO TO 75
      SUMX=0.
      SUMX2=0.
      DO 70 J=1, NDCOUNTS(I)
      X(J)=DEXP((DLOG(DCSTDSAM(I,J))-A0)/A1)
      SUMX=SUMX+X(J)
      SUMX2=SUMX2+X(J)**2
70  CONTINUE
      ARIMEAN=SUMX/NDCOUNTS(I)
      SDEV=DSQRT((SUMX2-SUMX**2/NDCOUNTS(I))/(NDCOUNTS(I)-1))
      WRITE(IPRTR,65)IDSUB(I),NDCOUNTS(I),(X(J),J=1,NDCOUNTS(I)),ARIMEAN
      C,SDEV
75  CONTINUE
      RETURN
      END
```

SOURCE LISTING FOR SUBROUTING XLNSQFT (Second Order Calibration)

SUBROUTINE XLNSQFT

C
C A LOG/LOG DATA TRANSFORMATION IS MADE, AND THE METHOD OF
C LEAST SQUARES IS USED TO CALCULATED THE COEFFICIENTS OF A
C SECOND ORDER STANDARD CURVE FROM THE TRANSFORMED DATA.
C THE UNTRANSFORMED CALIBRATION DATA SET, TOGETHER WITH
C THE ANTILOGS OF PREDICTED Y-VALUES, ARE PRINTED OUT.
C FINALLY, CONCENTRATIONS OF THE ANALYTE IN SAMPLES ARE
C PRINTED.

C
C VARIABLES ARE GROUPED AND TYPED, ARRAYS ARE
C DIMENSIONED, AND DCB'S ARE NAMED.
C

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
COMMON/BLK3/DCSTDSAM(60,10),NDCOUNTS(60),SUBCONC(60),IDSUB(60),IDC

CROW
DIMENSION CALSTD(60,10), IDSTD(60),STDCONC(60),X(10),NSCOUNTS(60)

DATA INPUT/105/IPRTR/108/

C
C CALIBRATION STANDARDS ARE SORTED OUT FROM SAMPLES, AND
C SUMMATIONS ARE PERFORMED.
C

SUMY=0.
SUMXY=0.
SUMX2Y=0.
SUMX=0.
SUMX2=0.
SUMX3=0.
SUMX4=0.
N=0
NROWS=0
DO 20 I=1, IDCROW
IF(SUBCONC(I).EQ.0.)GO TO 20
N=N+NDCOUNTS(I)
NROWS=NROWS+1
DO 10 J=1, NDCOUNTS(I)
SUMY=SUMY+DLOG(DCSTDSAM(I,J))
SUMXY=SUMXY+DLOG(SUBCONC(I))*DLOG(DCSTDSAM(I,J))
SUMX2Y=SUMX2Y+DLOG(DCSTDSAM(I,J))*(DLOG(SUBCONC(I)))**2
SUMX=SUMX+DLOG(SUBCONC(I))
SUMX2=SUMX2+(DLOG(SUBCONC(I)))**2
SUMX3=SUMX3+(DLOG(SUBCONC(I)))**3
SUMX4=SUMX4+(DLOG(SUBCONC(I)))**4
CALSTD(NROWS, J)=DCSTDSAM(I, J)

10 CONTINUE
IDSTD(NROWS)=IDSUB(I)
STDCONC(NROWS)=SUBCONC(I)
NSCOUNTS(NROWS)=NDCOUNTS(I)
20 CONTINUE

C
C
C

THE REGRESSION COEFFICIENTS ARE CALCULATED.

```
SYSTDET=N*(SUMX2*SUMX4-SUMX3**2)-SUMX*(SUMX*SUMX4-SUMX3*SUMX2)+SUM
CX2*(SUMX*SUMX3-SUMX2**2)
A0=(SUMY*(SUMX2*SUMX4-SUMX3**2)-SUMX*(SUMXY*SUMX4-SUMX3*SUMX2Y)+SU
CMX2*(SUMXY*SUMX3-SUMX2*SUMX2Y))/SYSTDET
A1=(N*(SUMXY*SUMX4-SUMX3*SUMX2Y)-SUMY*(SUMX*SUMX4-SUMX3*SUMX2)+SUM
CX2*(SUMX*SUMX2Y-SUMXY*SUMX2))/SYSTDET
A2=(N*(SUMX2*SUMX2Y-SUMXY*SUMX3)-SUMX*(SUMX*SUMX2Y-SUMXY*SUMX2)+SU
CMY*(SUMX*SUMX3-SUMX2**2))/SYSTDET
```

C
C
C

CALIBRATION DATA ARE PRINTED OUT.

```
WRITE(IPRTR,22)
22 FORMAT(///,16X,'CALIBRATION DATA')
WRITE(IPRTR,23)
23 FORMAT(16X,'-----',/)
WRITE(IPRTR,25)
25 FORMAT(2X,'STD',3X,'STD',T50,'PREDICTED')
WRITE(IPRTR,30)
30 FORMAT(1X,'IDENT',2X'CONC',5X,'SLOPE-CORRECTED COUNTS',12X,'COUNT
CS')
WRITE(IPRTR,35)
35 FORMAT(1X,'-----',2X,'-----',5X,'-----',12X,'-----
C-',/)
40 FORMAT(2X,A4,1X,F7.2,3X,NF5.0,T50,F5.0)
DO 45 I=1,NROWS
WRITE(IPRTR,40)IDSTD(I),STDCONC(I),NSCOUNTS(I),CALSTD(I,J),J=1,NS
CCOUNTS(I),DEXP(A0+A1*DLOG(STDCONC(I))+A2*(DLOG(STDCONC(I)))**2)
45 CONTINUE
WRITE(IPRTR,50)A0,A1,A2
50 FORMAT(//,1X,'A0=',F10.7,2X,'A1=',F10.7,2X,'A2=',F10.7)
```

C
C
C
C
C

SAMPLES ARE CULLED FROM CALIBRATION STANDARDS, AND
ANALYTE CONCENTRATIONS (WITH MEAN AND STANDARD
DEVIATION) ARE CALCULATED.

```
WRITE(IPRTR,52)
52 FORMAT(///,16X,'ANALYTICAL RESULTS')
WRITE(IPRTR,53)
53 FORMAT(16X,'-----',/)
WRITE(IPRTR,55)
55 FORMAT(2X,'IDENT',4X'ANALYTE CONCENTRATIONS')
WRITE(IPRTR,60)
60 FORMAT(2X,'-----',4X,'-----')
65 FORMAT(1X,A5,NF6.0,2X,'MEAN=',F7.2,2X,'SDEV=',F6.2)
CRTRN1=2.*A2
CRTRN2=DEXP(-A1/CRTRN1)
DO 75 I=1, IDCROW
IF(SUBCONC(I).NE.0.)GO TO 75
SUMX=0.
SUMX2=0.
```

```
DO 70 J=1, NDCOUNTS(I)
X(J)=DEXP((-A1+DSQRT(A1**2-4.*A2*(A0-DLOG(DCSTDSAM(I,J)))))/(2.*A2
C))
IF(CRTRN1.GT.0..AND.X(J).GT.CRTRN2)GO TO 67
IF(CRTRN1.LT.0..AND.X(J).LE.CRTRN2)GO TO 67
X(J)=DEXP((-A1-DSQRT(A1**2-4.*A2*(A0-DLOG(DCSTDSAM(I,J)))))/(2.*A2
C))
67 SUMX=SUMX+X(J)
SUMX2=SUMX2+X(J)**2
70 CONTINUE
ARIMEAN=SUMX/NDCOUNTS(I)
SDEV=DSQRT((SUMX2-SUMX**2/NDCOUNTS(I))/(NDCOUNTS(I)-1))
WRITE(IPRTR,65)IDSUB(I),NDCOUNTS(I),(X(J),J=1,NDCOUNTS(I)),ARIMEAN
C,SDEV
75 CONTINUE
RETURN
END
```

REPORT DOCUMENTATION PAGE	1. REPORT NO. WHOI-81-88	2.	3. Recipient's Accession No.
4. Title and Subtitle A COMPUTER PROGRAM TO PROCESS DATA FROM A DIRECT CURRENT PLASMA EMISSION SPECTROMETER		5. Report Date October 1981	6.
7. Author(s) Donald C. Bankston		8. Performing Organization Rept. No.	
9. Performing Organization Name and Address Woods Hole Oceanographic Institution Woods Hole, Massachusetts 02543		10. Project/Task/Work Unit No.	11. Contract(C) or Grant(G) No. (C)DE-A102-81EV10694 (G)OCE-81-19056
12. Sponsoring Organization Name and Address U.S. Department of Energy and National Science Foundation		13. Type of Report & Period Covered Technical	14.
15. Supplementary Notes This report should be cited as: Woods Hole Oceanog. Inst. Tech. Rept. WHOI-81-88.			
16. Abstract (Limit: 200 words) This program processes data from a Spectraspan IIB spectrometer retrofitted with a Spectrajet III plasma excitation source (Spectrametrics Inc., Andover, MA). The spectrometer is operated manually and can determine only one element at a time. Accordingly, the program treats data for each element individually. Two of the spectrometer's performance characteristics tend to change gradually during normal operations: average blank count and analytical sensitivity. The program compensates for these variations. For the program to process data from the spectrometer, a reagent blank and high standard (in this order) should be aspirated first; then the high standard and blank (in this order) should be run again after every six or fewer samples. The program, moreover, causes all data to be converted to logarithmic form before being used in computation. This makes possible the calculation of statistically valid confidence limits about predicted analyte concentrations. The program consists of a main section, DCPEOES, and three external subroutines: BASELINE, SENSITIV, and XLNSQFT. BASELINE compensates for any change in the average blank count, and SENSITIV corrects for any variation in analytical sensitivity. XLNSQFT governs all correlation and prediction functions and has two interchangeable versions: one for first-order; the other for second-order, correlation.			
17. Document Analysis a. Descriptors 1. Computer program 2. Direct current plasma emission spectrometry 3. Calibration curve b. Identifiers/Open-Ended Terms c. COSATI Field/Group			
18. Availability Statement		19. Security Class (This Report) Unclassified	21. No. of Pages 26
		20. Security Class (This Page)	22. Price

