1-1-1970

SICURV : a computer program to derive site index curves

Roger N. Baughman

D. L. Kulow

Follow this and additional works at: https://researchrepository.wvu.edu/wv_agricultural_and_forestry_experiment_station_bulletins

Digital Commons Citation

Baughman, Roger N. and Kulow, D. L., "SICURV : a computer program to derive site index curves" (1970). *West Virginia Agricultural and Forestry Experiment Station Bulletins*. 590T.
https://researchrepository.wvu.edu/wv_agricultural_and_forestry_experiment_station_bulletins/694

This Bulletin is brought to you for free and open access by the Davis College of Agriculture, Natural Resources And Design at The Research Repository @ WVU. It has been accepted for inclusion in West Virginia Agricultural and Forestry Experiment Station Bulletins by an authorized administrator of The Research Repository @ WVU. For more information, please contact ian.harmon@mail.wvu.edu.

SICURV

A Computer Program To Derive Site Index Curves

Bulletin 590T

March 1970

**West Virginia University
Agricultural Experiment Station**

THE AUTHORS

Roger N. Baughman was a Graduate Research Assistant at the time of this study. D. L. Kulow is Associate Forester.

ACKNOWLEDGMENT

The authors thank the West Virginia University Computer Center for donating the free use of the 7040 for debugging and testing this program.

DISCLAIMER

Although this program has been tested by its authors, no warranty, expressed or implied, is made by the authors, or by West Virginia University, as to the accuracy and functioning of the program and related program material, nor shall the fact of distribution constitute any such warranty, and no responsibility is assumed by the authors or West Virginia University in connection therewith.

West Virginia University
Agricultural Experiment Station
College of Agriculture and Forestry
A. H. VanLandingham, Director
Morgantown

SICURV

A Computer Program To Derive Site Index Curves

ROGER N. BAUGHMAN and D. L. KULOW

This bulletin describes the preparation and use of SICURV, a computer program which will derive and plot site index curves according to given characteristics. The program follows a strict statistical analysis in rejecting abnormal data and deriving regression equations. The program also provides diagnostic error messages which will facilitate the correction of errors in input data.

The program is written in FORTRAN IV computer language. It was debugged and tested on an IBM 7040/7044, 16/32K (IBSYS operating system). The program currently requires approximately 20,000 words of core storage, but may be easily modified for smaller computers.

The program also requires that the function sub-program AMIN1, AMAX1, ALOG, MOD, ABS, and EXP be internal to the computer. The program also suppresses error messages caused by numbers approaching zero (floating point traps) through the use of FPTTRA, a subprogram peculiar to the WVUCC. This may be removed and substitutions made, if necessary or desirable.

FUNCTION AND RATIONALE

By definition, the site index of an area is the average height that trees may be expected to attain at 50 years of age. With paired observations on tree height and age, a regression equation that expresses tree height as a function of age may be fitted to the data. The height of the curve at age 50 then represents the site index of the area.

The height of other site index curves may be determined from the initial height curve. If the height of the growth curve at age 50 is 62 feet, the height of a curve representing site index

70 is $70/62 = 1.129$ times as high as the initial growth curve at any age, or in general:

$$H_{ij} = ((SI_i/H50) - 1)h_j + h_j \quad (1)$$

where:

H_{ij} = height of the i th site index curve at the j th age,
 SI_i = the value of the i th site index curve at any age,
 h_j = height of the guide curve at the j th age.

Equation (1) assumes that there is no systematic variation in height growth as age increases. A measure of this variation, if it exists, may be made in the form of coefficient of variation or:

$$CV_j = (S_j) / (\bar{X}_j) \quad (2)$$

where:

CV_j = coefficient of variation at the j th age,
 S_j = standard deviation of height at the j th age,
 $\bar{X}_j$ = mean height at the j th age.

A regression equation to predict coefficient of variation would require a large number of observations if ages are considered individually. A smaller number of observations may be taken if the data are separated into age classes. If the variation in height growth does change significantly with age, Equation (1) may be expanded to:

$$H_{ij} = (((SI_i/H50) - 1) / CV_{50}) CV_j h_j + h_j \quad (3)$$

where:

CV_{50} = coefficient of variation at age 50,
 CV_j = coefficient of variation at the j th age.

SICURV develops site index curves according to this logic.

TESTING OF DATA

SICURV reads in control cards which specify the limits to be observed on different characteristics (see preparation of data) followed by paired observations of tree height and age. The data is then sorted into age class, which the user must specify, and the following tests are made:

(1) If less than three trees occur in an age class during the initial sort of that age class, the class is dropped from calculations.

(2) All observations of height in each age class are tested for normality by the Z-test. Any tree with an absolute Z-value greater than 2.5 is discarded from its age class. The mean and standard deviation are recalculated when a tree is rejected and testing continues. When tests have been completed for an age class, all observations rejected from that age class are listed, and those observations which remain in that age class are listed. After all data has been sorted into the range of age classes specified by the user, observations outside of the specified range are listed so that all observations may be accounted for.

(3) A check is then made on the number of observations remaining in each age class. If less than three observations remain in any age class, all processing is terminated. The user then must collect more data or change the range of age classes.

COEFFICIENT OF VARIATION

If the data are acceptable after the sorting procedure, the coefficient of variation for each age class is used to predict the coefficient of variation at any age. Four different equations are tested by the program. The general form of the equations is:

$$CV_j = a + b X_j \quad (4)$$

where:

CV_j = coefficient of variation at the j th age,

a = y-intercept of the equation,

b = slope of the equation,

X = a transformation of age as age, age^2 , $1/\text{age}^2$, $1/\text{age}^{1/2}$.

The transformation of age which leaves the smallest residual sums of squares in the regression is used by the program as the equation which best fits the data. An analysis of variance is then performed to test the significance of the regression coefficients of the selected equation. If both coefficients are significant, both are used. If a is non-significant, b is recalculated. If b is non-significant, or both are non-significant, then trivially all $CV_j = 1$.

If coefficient of variation is to be used in the derivation of the site index curves, it is solved over a specified range of ages by one-year intervals. The graph of the equation is displayed on the printer and the values on the graph are also listed.

THE GUIDE CURVE

Transformations are then made on all paired observations

which were not discarded during sorting and they are fitted to the equation.

$$\ln Ht_i = a + b/X_i, \text{ or} \quad (5)$$

$$Ht_i = e^{(a + b/X_i)} \quad (6)$$

where:

Ht_i = height of the guide curve at the i th age,

X_i = $1/\text{age}$ at the i th age,

e = the base of the napierian logarithm.

This equation is then solved for the same range of values as the equation for coefficient of variation. The graph of the values are displayed on the printer, and the values are also listed.

THE SITE INDEX CURVES

Equations (4) and (5) are combined to form Equation (3) and then solved for the range of ages and site indices specified by the user. The family of site index curves are displayed on the printer. The values used to plot the site index curves are also listed for more exact plotting by hand if desired.

It should be noted that when a selective process is used to collect data, the regression equation for coefficient of variation may often be of the form $CV = a + b/X^2$, or $CV = a + b/X^{1/2}$. Depending on the regression coefficients a and b , the equation for coefficient of variation may be the near inverse of the guide curve of height over age. In this event, the two equations will cancel each other and the site index curves will be nearly parallel as indicated by the data.

An example of a set of data which produced this result is listed in Appendix A¹. The resultant execution by SICURV on this data is given in Appendix C.

PREPARATION OF DATA

The paired observations of height and age may be arranged as the user desires or has already arranged. Three control cards are required preceding the data, and all specifications on these cards must be present. SICURV does not check for illogical entries on these control cards, so their validity is in the hands of the user.

¹Data taken from *Site Index Curves for Black Cherry in Northern West Virginia with an Associated Study of Variation in Specific Gravity with Site Index*, 1964, Unpublished Thesis, H. H. Heesch.

The first card of the input data is a specification control card entered under the format (I6,3F6.1,I3,5F6.1,F9.4).^{*} The specifications to be entered into these 11 fields are as follows:

cc	specifications
(1) 1- 6	The total number of paired observations of height and age.
(2) 7-12	The lower limit of the lowest age class into which the data is to be sorted. If the lowest age class is 10 years with an age class interval of 10 years, this value would be 5.5.
(3) 13-18	The age interval of each age class. From the above example, enter the number 10.
(4) 19-24	The upper bound of the oldest age class into which the data is to be sorted. The reasoning is the same as for specification (2).
(5) 25-27	The number of site index curves to be plotted. A maximum of 10 is allowed.
(6) 28-33	The site index of the lowest site index curve to be plotted.
(7) 34-39	The interval between each site index curve.
(8) 40-45	The site index of the highest site index curve to be plotted.
(9) 46-51	The lowest age over which the site index curves are to be plotted.
(10) 52-57	The highest age over which the site index curves are to be plotted. Values of height are calculated for each 1-year increment in age. The maximum allowable difference between specifications (9) and (10) is 110 years.
(11) 58-66	The F value used to test the regression coefficients for coefficient of variation. The value may be used at the alpha level desired with 2, n-1, degrees of freedom, where n is the number of age classes into which the data is to be sorted.

The second card provides the user with the means of labeling the output. Any pertinent information may be entered here from the card column 1 through 78.

^{*}For an explanation of FORMAT statements, see Anderson, D. M. **Computer Programming FORTRAN IV**, Appleton-Century-Crofts, 1966, or any other book on FORTRAN language.

The third card is a format card which specifies the location of the data on the input cards. The format card must begin with a left parenthesis mark in card column 1 and may continue to card column 72. In card column 80 of this card must be punched a code which specifies the order of the input data. If height is the first entry on each card, enter the number 1. If age precedes height, the field may be left blank.

The input data then follows this card. The number of paired observations indicated must be present, and they must agree in format and order as specified on the previous cards. An example of input data is shown in Appendix A.

ERROR MESSAGES

In addition to FORTRAN IV error messages, which may be encountered, SICURV provides the user with error messages which indicate insufficient data for a valid statistical analysis, or that abnormal data has been encountered and has been dropped. The messages and the action to be taken are as follows:

1. **LESS THAN THREE TREES IN THIS CLASS, CLASS DROPPED FROM CALCULATIONS.** SICURV has found that in the initial sort, the age class which is listed above this message does not contain sufficient observations for valid tests. The class may be eliminated by changing control cards, or more data must be collected in this age class. The cause of this message will terminate processing at the end of the sorting procedure.

2. **NO TREES IN THIS CLASS, CONTINUING TO THE NEXT CLASS.** SICURV has found no trees in the age class listed above this message. The procedure described above should be followed. Processing terminates at the end of the sorting procedure if this message is given.

3. **NUMBER OF TREES IN THIS CLASS NOW LESS THAN THREE DUE TO REJECTIONS BY Z-TEST, CLASS DROPPED FROM CALCULATIONS.** SICURV has found that after rejections by the Z-test, less than three trees remain in the age class listed above the message and the age class is dropped. Processing is terminated at the end of the sorting procedure if an age class is dropped, and appropriate steps should be taken to eliminate this condition.

4. **THIS TREE DOES NOT FIT THE NORMAL POPULATION DEFINED BY THIS AGE CLASS. AGE IS, HEIGHT IS, TEST STATISTIC IS** This represents the rejection

of an observation by SICURV due to the high value of absolute Z. The error is not critical to the program so long as three or more trees remain in the age class.

5. THE FOLLOWING TREES TOO LOW FOR ANY AGE CLASS. The ages of the trees listed below this message are too low for any specified age class and are therefore dropped from any further consideration. Processing will continue unaffected.

6. THE FOLLOWING TREE(S) TOO HIGH FOR ANY AGE CLASS. DROPPED FROM CALCULATIONS. Similar to message 5, but the trees are older than the upper bound of the oldest age class. Processing will continue after this message.

As indicated above, the data will be sorted as specified by the user before processing is halted due to errors. This should allow the user to eliminate all messages which will terminate processing before the next run. As execution continues after sorting other messages are given to indicate what function the computer is performing and the equations used to derive the site index curves.

APPENDIX A

Partial Input Data List

The control cards in this sample listing will instruct SICURV to separate data into age classes on 10-year intervals. The youngest age class is 20 years, the oldest age class is 80 years. SICURV is instructed to construct eight site index curves. The curves are to begin at site index 50 and continue to site index 120 at intervals of 10. The curves are to be plotted from ages 20 to 60. The data specify that an F value of 6.61 is to be used in the analysis of variance. The graphs are to be labeled with the title "Prunus Serotina — Black Cherry Data from Northern West Virginia-H. H. Heesch." The format specifications for the data are (2F6.1), height is the first variable recorded on each card, and there are 396 paired observations.

396 15.0 10.0 85.0 8 50.0 10.0 120.0 20.0 60.0 6.61 Prunus
Serotina — Black Cherry Data from Northern West Virginia-H.
H. Heesch (2F6.1). 1

62.0 41.0
64.0 41.0
70.0 41.0
71.0 41.0
73.0 41.0
75.0 41.0

79.0	41.0
81.0	41.0
70.0	42.0
76.0	42.0
89.0	42.0
90.0	42.0
92.0	42.0
74.0	43.0
76.0	43.0
76.0	43.0
79.0	43.0
85.0	43.0
91.0	43.0
60.0	44.0
63.0	44.0
65.0	44.0
67.0	44.0
67.0	44.0
75.0	44.0
76.0	44.0
77.0	44.0
79.0	44.0
74.0	35.0
77.0	35.0
77.0	35.0
81.0	35.0
58.0	36.0
85.0	36.0
86.0	36.0
60.0	37.0
66.0	37.0
71.0	37.0
79.0	37.0
81.0	37.0
81.0	37.0
56.0	38.0
63.0	38.0
67.0	38.0
78.0	38.0
87.0	38.0
87.0	38.0
63.0	40.0
71.0	40.0
78.0	40.0
81.0	40.0
90.0	40.0
96.0	40.0
62.0	41.0

\$IBSYS

APPENDIX B - SICURV PROGRAM LISTING (MAIN PROGRAM)

```

$IBJOB          MAP
$IBFTC RNB001  DECK
C      WRITTEN BY ROGER BAUGHMAN
C      WEST VIRGINIA UNIVERSITY
C      DETERMINATION OF SITE INDEX CURVES
      DIMENSION D(50),PLOC(150,10),PLSI(150,10),PLAC(150,1),PLGC(150,1),
      1PLCV(150,1),H(10),NPNTS(10),AS(50),B(50),G(50),CV(50),ANM(50),HT(2
      1000),AG(2000),TLOW(1500),TOHIGH(1500),RAGE(2000), ELGHT(2000),IAG
      4(500)
      DIMENSION RNB(12)
      IOVFLO=0
      IUNFLO=0
      COMMON/FPITRA/IUNFLO,IOVFLO,IPRINT,NOLIM
      IPRINT=-1
      NOLIM=-1
5001 FORMAT (I6,F6.1,F6.1,F6.1,I3,F6.1,4(F6.1),F9.4)
5002 FORMAT (13(A6))
5003 FORMAT (1H0,40X,50HSTATISTICAL ANALYSIS SUGGESTED BY DR. DON L. KU
      1LOW)
5004 FORMAT(2(F6.1))
5005 FORMAT (12A6,7X,I1)
5006 FORMAT (1H0,55X,20HDIVISION OF FORESTRY)
5007 FORMAT (1H0,53X,24HWEST VIRGINIA UNIVERSITY)
5008 FORMAT (1H0,52X,25HMORGANTOWN, WEST VIRGINIA)
5999 FORMAT(1H0,20(6H*****))
5998 FORMAT (1H0, 16HAGE CLASS NUMBER,I3,13H EXTENDS FROM,F6.1,3H TO,F6
      1.1)
5997 FORMAT (1H0,66HLESS THAN TWO TREES IN THIS CLASS,CLASS DROPPED FRO
      1M CALCULATIONS.)
5996 FORMAT (1H0,70HTHE FOLLOWING TREE(S) HAVE BEEN DROPPED FROM CALCUL
      1ATIONS AGE HEIGHT)
5995 FORMAT (1H0,55HNO TREES OCCUR IN THIS CLASS. CONTINUING TO NEXT CL
      1ASS.)
5994 FORMAT (1H0,74HTHIS TREE DOES NOT FIT THE NORMAL POPULATION DEFINE
      1D BY THIS CLASS. AGE IS,F6.1,10H HEIGHT IS,F6.1,18H TEST STATISTIC
      1 IS,F6.2)
5993 FORMAT (1H ,31HTREE DROPPED FROM CALCULATIONS.)
5992 FORMAT (1H0,106HNUMBER OF TREES IN THIS CLASS NOW LESS THAN 3 DUE
      1TO REJECTIONS BY ZTEST, CLASS DROPPED FROM CALCULATIONS.)
5991 FORMAT (1H ,50HTREES DROPPED FROM CALCULATIONS ARE, AGE HEIGHT.)
5990 FORMAT (1H ,37X,F6.1,1X,F6.1)
5989 FORMAT (1H ,56X,F6.1,2X,F6.1)
5988 FORMAT (1H ,13HTHE FOLLOWING,I4,34H TREES ARE INCLUDED IN THIS CLA
      1SS.)
5987 FORMAT (1H ,20X,8HTREE AGE,3X,11HTREE HEIGHT)
5986 FORMAT (1H ,23X,F6.1,6X,F6.1)
5985 FORMAT (1H0,6X,8HSUM AGES,3X,11HSUM AGE SQD,5X,4HMEAN,4X,5HSTDEV,4
      1X,9HCOEFF VAR,4X,9HSUM OF HT,4X,11HSUM OF HTSQ,4X,10HST.DEV,HT.)
5984 FORMAT (1H ,5X,F10.1,2X,F11.1,3X,F6.1,2X,F10.4,3X,F6.2,4X,F10.1,4X
      1,F11.1,4X,F10.4)
5983 FORMAT (1H1,113HLESS THAN 3 TREES IN ONE OR MORE CLASSES ABOVE,IN

```

```

1SUFFICIENT DATA FOR VALID STATISTICAL ANALYSIS--JOB TERMINATED.)
5979 FORMAT (1H ,12HSPECIES IS. ,13(A6))
5967 FORMAT (1H0,1H ////67HAFTER REGRESSION AND ANOV,THE FOLLOWING EQUA
ITION REMAINS VALID- CV=.F9.4,1H+.F9.4.22HX, X A FUNCTION OF AGE)
5966 FORMAT (1H1,75HTHE FOLLOWING TREE(S) TOO LOW FOR ANY AGE CLASS, DR
OPPED FROM CALCULATIONS.)
5965 FORMAT (1H ,6X,F6.1)
5964 FORMAT (1H0,76HTHE FOLLOWING TREE(S) TOO HIGH FOR ANY AGE CLASS, D
ROPPED FROM CALCULATIONS.)
5963 FORMAT (1H1,38X,42HGRAPH OF COEFFICIENT OF VARIATION OVER AGE)
5962 FORMAT (1H0,21X,13(A6))
5961 FORMAT (1H1,66HTHE FOLLOWING VALUES ARE PLOTTED ON THE GRAPH ABOVE
1. COEF.VAR. AGE)
5960 FORMAT (1H ,54X,F6.2,4X,F6.1)
5959 FORMAT (1H1,37HGUIDE CURVE OF TOTAL HEIGHT OVER AGE.)
5958 FORMAT (1H0,13(A6))
5957 FORMAT (1H1,54HTHE FOLLOWING ARRAYS WERE USED TO PLOT THE GUIDE CU
IRVE)
5956 FORMAT (1H0,17X,3HAGE,11X,12HTOTAL HEIGHT)
5955 FORMAT (1H ,15X,F6.1,13X,F6.1)
5954 FORMAT (1H1,22HSITE INDEX CURVES FOR.)
5953 FORMAT (1H0,13(A6))
5952 FORMAT(1H0,46X,10HSITE INDEX)
5951 FORMAT(1H0,5X,3HAGE,4X,F6.1,9(3X,F6.1))
5950 FORMAT(1H ,11(3H*-.F6.1),3H*-.*)
5949 FORMAT (1H0,68HCOEFF.VAR. N.S. AT USER SPECIFIED ALPHA LEVEL, CONS
IDERED EQUAL TO 1)
5948 FORMAT (1H0,63HIN THE MATRIX ABOVE, X=AGE, Y=HEIGHT, A=NUMBER OF O
BSERVATIONS.)
5947 FORMAT (1H0,11X,1HA,8X,1HX,13X,1HY)
5946 FORMAT (1H0,1HA,5X,F7.1,2X,F11.5,3X,F11.5)
5945 FORMAT (1H0,1HX,14X,F11.5,3X,F11.5)
5944 FORMAT (1H0,1HY,28X,F11.5)
5943 FORMAT (1H0,3HRES,26X,F11.5)
5942 FORMAT (1H0,49HGUIDE CURVE OF TOTAL HEIGHT OVER AGE. ----- HT=.F
19.5,1H+.F9.5,4H/AGE)
5941 FORMAT (1H0,7(6H*****))
5940 FORMAT (1H1,61HTHE FOLLOWING ARRAYS WERE USED TO PLOT THE SITE IND
EX CURVES.)
5939 FORMAT (1H ,9X,3HAGE)
5938 FORMAT (1H0,3HRES,43X,F15.1)
5937 FORMAT (1H ,11(6H*****))
5936 FORMAT (1H0,106HIN THE MATRIX ABOVE, A=NUMBER OF OBSERVATIONS, X=M
EAN AGE, Z=MEAN AGE SQUARED, Y=COEFFICIENT OF VARIATION.)
5935 FORMAT (1H1,36HERROR IN PROGRAMMING--JOB TERMINATED)
C
  READ IN CONTROL VALUES AND DATA.
  READ(5,5001)INCNT,ALB,AGINT,AUB,NFNS,SIMIN,SINT,SIMAX,SLOW,SIHIG
  1H,F
  READ(5,5002)(ANM(LO),LO=1,13)
  READ(5,5005)(RNB(I),I=1,12),IRC
  IF(IRC.EQ.1)GO TO 99
  DO 98 I=1,INCNT
  98 READ(5,RNB)AG(I),HT(I)
  GO TO 101
  99 DO 100 I=1,INCNT

```

```

100 READ(5,RNB)HT(I),AG(I)
C   INITIALIZE COUNTERS FOR SORTING DATA INTO AGE CLASSES.
101 SORT=ALB+AGINT
    KK=0
    MM=0
    JL=0
    NOGO=1
    ALT=ALB
    1 SOHT=0.
      SOHTSQ=0.
      JJ=0
      P=0.
      L=0
      SUMAGE=0.
      SAGES=0.
C   BEGIN SORTING.
    DO 200 I=1,INCNT
      IF(AG(I).LE.SORT)GO TO 201
    70 GO TO 200
    201 IF(AG(I).GT.ALT)GO TO 210
    209 GO TO 200
    210 P=P+1.
        L=L+1
C   ACCUMULATE SUMS AND SUMS OF SQUARES FOR AGE AND HEIGHT.
        SUMAGE=SUMAGE+AG(I)
        SAGES=SAGES+AG(I)**2
        SOHT=SOHT+HT(I)
        SOHTSQ=SOHTSQ+HT(I)**2
        IAG(L)=I
    200 CONTINUE
        KK=KK+1
        WRITE(6,5999)
        ALCL=SORT-AGINT
        WRITE(6,5998) KK,ALCL,SORT
        IF(L.GE.3) GO TO 225
C   REJECT ANY CLASS WITH LESS THAN 3 OBSERVATIONS.
    211 NOGO=2
        IF(L.EQ.0)GO TO 226
    202 WRITE(6,5997)
        WRITE(6,5996)
        JK=IAG(1)
    203 WRITE(6,5989) AG(JK),HT(JK)
        AG(JK)=0.
        IF(L.EQ. JK)GO TO 275
        JK=IAG(2)
        GO TO 203
    226 WRITE(6,5995)
        GO TO 275
C   BEGIN Z-TEST FOR ABNORMAL DATA.
    225 STDEV2=((SOHTSQ-(SOHT**2)/P)/(P-1.))**.5
        HTMEAN=SOHT/P
        I=1
        IT=IAG(I)
    230 ZTEST=(HT(IT)-HTMEAN)/STDEV2
        IF(ZTEST.GT.2.5)GO TO 409

```

```

69 IF((ZTEST).LT.(-2.5))GO TO 409
235 IF(I.EQ.L)GO TO 410
    I=I+1
    IT=IAG(I)
    GO TO 230
C    PRINT OUT REJECTED DATA AND REDUCE SUMS AND SUMS OF SQUARES.
409 WRITE(6,5994)AG(IT),HT(IT),ZTEST
    WRITE(6,5993)
    JJ=JJ+1
    P=P-1.
    SUMAGE=SUMAGE-AG(IT)
    SAGES=SAGES-AG(IT)**2
    SOHT=SOHT-HT(IT)
    SOHTSQ=SOHTSQ-HT(IT)**2
    HTMEAN=SOHT/P
    STDEV2=((SOHTSQ-(SOHT**2)/P)/(P-1.))**.5
    AG(IT)=0.
    GO TO 235
410 IMIN=L-JJ
C    SECOND REJECTION OF AN AGE CLASS WITH LESS THAN THREE TREES.
    IF(IMIN.GE.3)GO TO 420
416 WRITE(6,5992)
    WRITE(6,5991)
    I=1
    NOGO=2
    IM=IAG(I)
418 IF(AG(IM).EQ.0.)GO TO 419
417 WRITE(6,5990)AG(IM),HT(IM)
    AG(IM)=0.
419 IF(I.EQ.L)GO TO 275
    I=I+1
    IM=IAG(I)
    GO TO 418
C    CALCULATE AGE CLASS STATISTICS.
420 AMEAN=SUMAGE/P
    G(KK)=AMEAN
    STDEV=((SAGES-(SUMAGE**2)/P)/(P-1.))**.5
    CV(KK)=STDEV2/HTMEAN*100.
    INBR=L-JJ
    WRITE(6,5988)INBR
    WRITE(6,5987)
    I=1
422 NN=IAG(I)
    IF(AG(NN).NE.0.)GO TO 450
444 I=I+1
    IF(I.GT.L) GO TO 475
445 GO TO 422
C    PRINT TREES IN CURRENT AGE CLASS, AND STATISTICS OF THAT CLASS.
450 WRITE(6,5986) AG(NN),HT(NN)
    GO TO 444
475 WRITE(6,5985)
    WRITE(6,5984) SUMAGE,SAGES,AMEAN,STDEV,CV(KK),SOHT,SOHTSQ,STDEV2
476 SORT=AGINT
    ALT=AGINT
500 IF(SORT.LE.AUB) GO TO 1

```



```

501 DO 300 I=1,INCNT
C   REJECT TREES TOO OLD FOR ANY AGE CLASS.
    IF(AG(I).LT.AUB)GO TO 300
    JL=JL+1
    TOHIGH(JL)=AG(I)
    AG(I)=0.
300 CONTINUE
    DO 299 I=1,INCNT
    IF(AG(I).EQ.0.)GO TO 299
C   REJECT TREES TOO YOUNG FOR ANY AGE CLASS.
    IF(AG(I).GT.ALB)GO TO 299
    MM=MM+1
    TOLOW(MM)=AG(I)
    AG(I)=0.
299 CONTINUE
    IF(MM.EQ.0)GO TO 610
549 WRITE(6,5966)
    WRITE(6,5939)
    DO 609 I=1,MM
609 WRITE(6,5965) TOLOW(I)
610 IF(JL.EQ.0) GO TO 29
    33 WRITE(6,5964)
    WRITE(6,5939)
    DO 611 I=1,JL
611 WRITE(6,5965) TOHIGH(I)
    29 GO TO (301,303),NOGO
303 WRITE(6,5983)
    GO TO 9090
C   TRANSFORM MEAN AGE FOR EACH AGE CLASS.
301 DO 302 I=1,KK
    AS(I)=G(I)**2
    B(I)=1./AS(I)
302 D(I)=1./G(I)**.5
    P=FLOAT(KK)
C   SUMS, SUMS OF SQUARES AND CROSS-PRODUCTS OF ABOVE TRANSFORMATIONS.
    CVB=SCP(CV,B,KK)
    BSUM=SOMS(B,KK)
    BS=SOS(B,KK)
    SUCVS=SOMS(CV,KK)
    SUAS=SOMS(AS,KK)
    SUMAG=SOMS(G,KK)
    SURAS=SOMS(D,KK)
    CVS=SOS(CV,KK)
    CVX=SCP(CV,G,KK)
    CVXS=SCP(CV,AS,KK)
    CVRXS=SCP(CV,D,KK)
    SGS=SOS(G,KK)
    SASG=SCP(AS,G,KK)
    SASAS=SOS(AS,KK)
    SRARA=SOS(D,KK)
    ICURV=1
    CALL LIREG(P,SURAS,SUCVS,SRARA,CVRXS,CVS,A5,A6,ICURV,RES1)
    ICURV=2
    CALL LIREG(P,SUAS,SUCVS,SASAS,CVXS,CVS,A3,A4,ICURV,RES2)
    ICURV=3

```

```

CALL LIREG(P,SUMAG,SUCVS,SGS,CVX,CVS,A1,A2,ICURV,RES3)
ICURV=4
CALL LIREG(P,BSUM,SUCVS,BS,CVB,CVS,A7,A8,ICURV,RES4)
GDCRV=AMIN1(RES1,RES2,RES3,RES4)
IF(GDCRV.EQ.RES1)GO TO 8866
IF(GDCRV.EQ.RES2)GO TO 8865
IF(GDCRV.EQ.RES3)GO TO 8864
IF(GDCRV.EQ.RES4)GO TO 8867
C
EQUATION IS CV=A + B*X, X=AGE.
8864 ICURV=3
CALL LIREG(P,SUMAG,SUCVS,SGS,CVX,CVS,A1,A2,ICURV)
CALL VARAN(P,SUCVS,A2,CVRXS,SUMAG,CVS,F,IRES)
IF(IRES.NE.1)GO TO 8870
C1=A1
C2=A2
GO TO 560
8870 IF(IRES.NE.2)GO TO 565
C1=0.
C2=CVX/SGS
GO TO 560
C
EQUATION IS CV=A + B*X, X=AGE**2.
8865 ICURV=2
CALL LIREG(P,SUAS,SUCVS,SASAS,CVXS,CVS,A3,A4,ICURV)
CALL VARAN(P,SUCVS,A4,CVXS,SUAS,CVS,F,IRES)
IF(IRES.NE.1)GO TO 8871
C1=A3
C2=A4
GO TO 560
8871 IF(IRES.NE.2)GO TO 565
C1=0.
C2=CVXS/SASAS
GO TO 560
C
EQUATION IS CV=A + B*X, X=AGE**-1/2.
8866 ICURV=1
CALL LIREG(P,SURAS,SUCVS,SRARA, CVRXS,CVS,A5,A6,ICURV)
CALL VARAN(P,SUCVS,A6,CVRXS,SURAS,CVS,F,IRES)
IF(IRES.NE.1)GO TO 8872
C1=A5
C2=A6
GO TO 560
8872 IF(IRES.NE.2)GO TO 565
C1=0.
C2=CVRXS/SRARA
GO TO 560
C
EQUATION IS CV=A + B*X, X=AGE**1/2.
8867 ICURV=4
CALL LIREG(P,BSUM,SUCVS,BS,CVB,CVS,A7,A8,ICURV)
CALL VARAN(P,SUCVS,A8,CVB,BSUM,CVS,F,IRES)
IF(IRES.NE.1)GO TO 8873
C1=A7
C2=A8
GO TO 560
8873 IF(IRES.NE.2)GO TO 565
C1=0.
C2=CVB/BS

```

```

560 WRITE(6,5967)C1,C2
C  CALCULATE VALUES OF COEFFICIENT OF VARIATION AND PLOT BY MPLOTS.
620 PLCV(I,1)=66.
    NSBVL=10
    PLAC(I,1)=100.
    NPNTS(1)=IFIX((AUB-ALB)-AGINT)
    CALL MPLOTS(0,2,PLAC,PLCV,NPNTS,NSBVL,10)
    PLAC(I,1)=ALB+AGINT/2.
    I=1
    GO TO (4699,2699,699,3699),ICURV
699  PLCV(I,1)=C1+C2*(PLAC(I,1))
    IF(PLAC(I,1).EQ.AUB-AGINT/2.)GO TO 700
650  IF(PLAC(I,1).NE.50.)GO TO 679
678  CV50=PLCV(I,1)
679  IF(PLAC(I,1).NE.SILOW)GO TO 680
681  IO=I
680  I=I+1
    PLAC(I,1)=PLAC(I-1,1)+1.
    GO TO 699
2699 CONE=PLAC(I,1)**2
    PLCV(I,1)=C1+C2*CONE
    IF(PLAC(I,1).EQ.AUB-AGINT/2.)GO TO 700
C  HOLD VALUE OF COEFFICIENT OF VARIATION WHEN AGE = 50 YEARS.
2650 IF(PLAC(I,1).NE.50.)GO TO 2679
2678 CV50=PLCV(I,1)
C  HOLD VALUE OF COEFFICIENT OF VARIATION AT LOWEST SITE INDEX AGE.
2679 IF(PLAC(I,1).NE.SILOW)GO TO 2680
2681 IO=I
2680 I=I+1
    PLAC(I,1)=PLAC(I-1,1)+1.
    GO TO 2699
3699 CONE=1./(PLAC(I,1)**2)
    PLCV(I,1)=C1+C2*CONE
    IF(PLAC(I,1).EQ.AUB-AGINT/2.)GO TO 700
3650 IF(PLAC(I,1).NE.50.)GO TO 3679
3678 CV50=PLCV(I,1)
3679 IF(PLAC(I,1).NE.SILOW)GO TO 3680
3681 IO=I
3680 I=I+1
    PLAC(I,1)=PLAC(I-1,1)+1.
    GO TO 3699
4699 CONE=1./(PLAC(I,1)**.5)
    PLCV(I,1)=C1+C2*CONE
    IF(PLAC(I,1).EQ.AUB-AGINT/2.)GO TO 700
4650 IF(PLAC(I,1).NE.50.)GO TO 4679
4678 CV50=PLCV(I,1)
4679 IF(PLAC(I,1).NE.SILOW)GO TO 4680
4681 IO=I
4680 I=I+1
    PLAC(I,1)=PLAC(I-1,1)+1.
    GO TO 4699
700  WRITE(6,5963)
    WRITE(6,5962)(ANM(LO),LO=1,13)
    NFCV=1
    CALL MPLOTS(1,NFCV,PLAC,PLCV,NPNTS,6)

```



```

        WRITE(6,5961)
        DO 701 J=1,I
C      PRINT ARRAYS OF COEFFICIENT OF VARIATION REGRESSION.
701    WRITE(6,5960) PLCV(J,1),PLAC(J,1)
        GO TO 750
565    WRITE(6,5949)
        PLAC(1,1)=ALB+AGINT/2.
        DO 566 I=1,120
            IF(PLAC(I,1).NE.SILOW)GO TO 567
568    IO=I
567    PLAC(I+1,1)=PLAC(I,1)+1.
C      COEFFICIENT OF VARIATION NON-SIG. CONSIDERED EQUAL TO 1.00
566    PLCV(I,1)=1.
        CV50=1.
750    PP=0.
        I=1
        J=0
751    IF(AG(I).NE.0.)GO TO 760
752    IF(I.EQ.INCNT)GO TO 800
753    I=I+1
        GO TO 751
C      TRANSFORM HEIGHT AND AGE OF ALL VALID DATA FOR GUIDE CURVE.
760    J=J+1
        PP=PP+1.
        RAGE(J)=1./AG(I)
        ELGHT(J)=ALOG(HT(I))
        IF(I.EQ.INCNT)GO TO 800
        I=I+1
        GO TO 751
C      CALCULATE NORMAL REGRESSION MATRIX AND REGRESSION CONSTANTS.
800    SRAGE=SOMS(RAGE,J)
        SLHT=SOMS(ELGHT,J)
        SRAGES=SOS(RAGE,J)
        SLHTS=SOS(ELGHT,J)
        SRALHT=SCP(RAGE,ELGHT,J)
        SRASRA=SRAGES-(SRAGE**2)/PP
        SAXHRA=SRALHT-(SRAGE*SLHT)/PP
        SHTSRA=SLHTS-(SLHT**2)/PP
        GCRES=SHTSRA-(SAXHRA**2)/SRASRA
        GC2=SAXHRA/SRASRA
        GC1=(SLHT-GC2*SRAGE)/PP
C      PRINT REGRESSION MATRIX FOR GUIDE CURVE.
        WRITE(6,5942)GC1,GC2
        WRITE(6,5979)(ANM(LO),LO=1,13)
        WRITE(6,5947)
        WRITE(6,5941)
        WRITE(6,5946)PP,SRAGE,SLHT
        WRITE(6,5941)
        WRITE(6,5945)SRAGES,SRALHT
        WRITE(6,5941)
        WRITE(6,5944)SLHTS
        WRITE(6,5941)
        WRITE(6,5941)
        WRITE(6,5945)SRASRA,SAXHRA
        WRITE(6,5941)

```

```

WRITE(6,5944)SHTSRA
WRITE(6,5941)
WRITE(6,5941)
WRITE(6,5943)GCRES
WRITE(6,5948)
WRITE(6,5942)
PLGC(1,1)=66.
PLAC(1,1)=100.
NSBVL=10
CALL MPLOTS(0,2,PLAC,PLGC,NSBVL,10)
I=1
PLAC(I,1)=(ALB+AGINT/2.)
C   CALCULATE GUIDE CURVE VALUES AND PLOT THESE VALUES.
850 PLGC(I,1)=EXP(GC1+GC2/PLAC(I,1))
    IF(PLAC(I,1).EQ.(AUB-AGINT/2.))GO TO 900
C   HOLD THE GUIDE CURVE VALUE WHEN AGE = 50 YEARS.
851 IF(PLAC(I,1).NE.50.)GO TO 867
852 GCHT=PLGC(I,1)
867 IF(PLAC(I,1).NE.SILOW) GO TO 899
    JO=I
899 I=I+1
    PLAC(I,1)=PLAC(I-1,1)+1.
    GO TO 850
900 NONS=1
    WRITE(6,5959)
    WRITE(6,5958)(ANM(LO),LO=1,13)
    NPNTS(1)=((AUB-ALB)-AGINT)
    CALL MPLOTS(1,NONS,PLAC,PLGC,NPNTS,6)
    WRITE(6,5957)
    WRITE(6,5956)
    DO 901 J=1,I
C   PRINT OUT GUIDE CURVE VALUES.
901 WRITE(6,5955)PLAC(J,1),PLGC(J,1)
    PLSI(1,1)=66.
    PLOC(1,1)=100.
    NPNTS(1)=IFIX((SIHIGH-SILOW)+1.)
    DO 910 I=1,10
910 NPNTS(I)=NPNTS(1)
    CALL MPLOTS(0,2,PLOC,PLSI,NSBVL,10)
    CV50=CV50/100.
    ACON=((SIMIN/GCHT)-1.)/CV50
    KK=1
    SIHOLD=SIMIN
    JHOLD=IO
    IHOLD=JO
    JJ=1
C   CALCULATE HEIGHT OF SITE INDEX CURVES.
905 PLSI(KK,JJ)=ACON*(PLCV(10,1)/100.)*PLGC(JO,1)+PLGC(JO,1)
    IF(KK.EQ.NPNTS(JJ))GO TO 920
906 KK=KK+1
    JO=JO+1
    IO=IO+1
    GO TO 905
920 IF(JJ.EQ.NFNS)GO TO 9999
921 JJ=JJ+1

```

```

SIMIN=SIMIN+SIINT
JO=JHOLD
IO=JHOLD
ACON=((SIMIN/GCHT)-1.)/CV50
KK=1
GO TO 905
C   CALCULATE AGE ARRAYS CORRESPONDING TO THE ABOVE CURVES.
9999 PLOC(1.1)=SILOW
NP=NPNTS(1)
DO 8808 L=2,NP
8808 PLOC(L.1)=PLOC(L-1.1)+1.
DO 8809 K=1,NP
DO 8809 N=2,10
8809 PLOC(K.N)=PLOC(K.1)
WRITE(6.5954)
WRITE(6.5953)(ANM(LO),LO=1,13)
C   CALL MPLOTS TO PLOT THE SITE INDEX CURVES.
CALL MPLOTS(1,NFNS,PLOC,PLSI,NPNTS,6)
WRITE(6.5940)
WRITE(6.5952)
I=1
H(I)=SIHOLD
980 IF(H(I).EQ.SIMAX)GO TO 1001
981 I=I+1
H(I)=H(I-1)+SIINT
GO TO 980
1001 WRITE(6.5951) (H(J),J=1,I)
IF(NFNS.EQ.10)GO TO 1003
1004 JJ=JJ+1
KK=1
C   CALCULATE O VALUES IF NFNS LESS THAN 10.
1010 PLSI(KK,JJ)=0.
1005 IF(KK.EQ.NPNTS(1))GO TO 1007
1006 KK=KK+1
GO TO 1010
1007 IF(JJ.EQ.10)GO TO 1003
1008 JJ=JJ+1
KK=1
GO TO 1010
DO 1002 I=1,KK
C   PRINT OUT ACTUAL VALUES PLOTTED.
1002 WRITE(6.5950)PLOC(I.1),(PLSI(I.K),K=1,10)
WRITE(6.5003)
WRITE(6.5006)
WRITE(6.5007)
WRITE(6.5008)
9090 STOP
END

```

APPENDIX B - CONTINUED - SICURY PROGRAM LISTING (FUNCTION SUBPROGRA

```

$IBFTC P2      DECK
      FUNCTION SOS(A,MN)
      DIMENSION A(2000)
C      FIND SUMS OF SQUARES OF THE VALUES PASSED.
      SOS=0.
      DO 2000 I=1,MN
2000  SOS=SOS+A(I)**2
      RETURN
      END
$IBFTC P3      DECK
      FUNCTION SOMS(Z,NL)
      DIMENSION Z(2000)
C      FIND THE SUM OF ALL VALUES PASSED.
      SOMS=0.
      DO 2001 I=1,NL
2001  SOMS=SOMS+Z(I)
      RETURN
      END
$IBFTC P4      DECK
      FUNCTION SCP(FA,FB,IJ)
      DIMENSION FA(2000),FB(2000)
C      FIND THE SUM OF CROSS-PRODUCTS OF ALL VALUES PASSED.
      SCP=0.
      DO 2002 I=1,IJ
2002  SCP=SCP+(FA(I)*FB(I))
      RETURN
      END

```

APPENDIX B - CONTINUED - SICURV PROGRAM LISTING (SUBROUTINE SUBPROG)

```

$1BFTC P6      DECK
      SUBROUTINE LIREG(RP,XS,YS,XSS,XSYS,YSS,CV1,CV2,ICURV,CVRES1)
      8092 FORMAT (1H .53HREGRESSION EQUATION IN THE FORM-CV=A+BX, X=AGE**1/
12.)
      8384 FORMAT (1H0.21(2H-).1H-)
      8082 FORMAT (1H .47HREGRESSION EQUATION IN THE FORM-CV=A+BX, X=AGE.)
      8083 FORMAT (1H .20HCOEFFICIENTS ARE, A=.F10.5,1X.2HB=.F10.5)
      8081 FORMAT (1H1.42HNORMAL FORWARD SOLUTION REGRESSION MATRIX.)
      8182 FORMAT (1H .55HREGRESSION EQUATION IN THE FORM-CV=A+BX, X=AGE SQUA
1RED.)
      8282 FORMAT (1H .53HREGRESSION EQUATION IN THE FORM-CV=A+BX, X=AGE**-1/
12.)
      8380 FORMAT (1H0.5X.1HA.14X.1HX.17X.1HY)
      8481 FORMAT (1H0.1HA.F7.1.3X.F15.4.3X.F15.4)
      8381 FORMAT (1H0.1HX.11X.F15.4.3X.F15.4)
      8382 FORMAT (1H0.1HY.29X.F15.4)
      8055 FORMAT (1H0.3HRES.25X.F15.4)
      SXSRA=XSS-((XS**2)/RP)
      SXYRA=XSYS-((XS*YS)/RP)
      SYSRA=YSS-((YS**2)/RP)
      CVRES1=SYSRA-((SXYRA**2)/SXSRA)
C      CALCULATE REGRESSION EQUATION COEFFICIENTS.
      CV2=SXYRA/SXSRA
      CV1=(YS-CV2*XS)/RP
C      PRINT OUT MATRIX AND EQUATION.
      WRITE(6,8081)
      IF(ICURV.NE.3)GO TO 8080
C      EQUATION IS CV=A+BX, X=AGE.
      8181 WRITE(6,8082)
      WRITE(6,8083)CV1,CV2
      GO TO 8888
      8080 IF(ICURV.NE.2)GO TO 8090
C      EQUATION IS CV=A+BX, X=AGE**2.
      WRITE(6,8182)
      WRITE(6,8083)CV1,CV2
      GO TO 8888
      8090 IF(ICURV.NE.1)GO TO 8091
C      EQUATION IS CV=A+BX, X=AGE**-1/2.
      WRITE(6,8282)
      WRITE(6,8083)CV1,CV2
      GO TO 8888
C      EQUATION IS CV=A+BX, X=AGE**1/2.
      8091 WRITE(6,8092)
      WRITE(6,8083)CV1,CV2
      8888 WRITE(6,8380)
C      PRINT OUT MATRIX AND RETURN.
      WRITE(6,8384)
      WRITE(6,8384)
      WRITE(6,8481)RP,XS,YS
      WRITE(6,8381)XSS,XSYS
      WRITE(6,8382)YSS

```

```

WRITE(6,8384)
WRITE(6,8384)
WRITE(6,8381)SXSRA, SXYRA
WRITE(6,8384)
WRITE(6,8382)SYSRA
WRITE(6,8384)
WRITE(6,8384)
WRITE(6,8055)CVRES1
RETURN
END
$IBFTC P7      DECK
SUBROUTINE VARAN(V,V1,CAT,V2,V3,V4,F,IRES)
6000 FORMAT (1H1,5HANO.V)
6001 FORMAT (1H0,6HSOURCE,3X,2HDF,7X,6HS OF S,5X,8HMEAN SQS,7X,2HFC,7X,
12HFT)
6002 FORMAT (1H ,11(5H*****))
6003 FORMAT (1H0,12HMEAN      1. ,F12.0,1X,F12.0,2X,F8.4,2X,F8.4)
6004 FORMAT (1H0,12HIND VAR   1. ,F12.0,1X,F12.0,2X,F8.4,2X,F8.4)
6005 FORMAT (1H0,8HRESIDUAL,F4.0,1X,F12.0,1X,F12.0,1X)
6006 FORMAT (1H0,5HTOTAL,3X,F4.0,1X,F12.0,1X,F12.0)
C   CALCULATE SUMS OF SQUARES AND MEAN SQUARES.
    ASS=(V1**2)/V
    VARSS=CAT*(V2-(V3*V1)/V)
    RSS=V4-(ASS+VARSS)
    RDF=V-2.
    RMS=RSS/RDF
C   CALCULATE F-VALUES.
    F1=ASS/RMS
    F2=VARSS/RMS
C   PRINT OUT ANALYSIS.
    WRITE(6,6000)
    WRITE(6,6001)
    WRITE(6,6002)
    WRITE(6,6003)ASS,ASS,F1,F
    WRITE(6,6002)
    WRITE(6,6004)VARSS,VARSS,F2,F
    WRITE(6,6002)
    WRITE(6,6005)RDF,RSS,RMS
    WRITE(6,6002)
    WRITE(6,6002)
    WRITE(6,6006)V,V4
C   DETERMINE LEVEL OF SIGNIFICANCE.
    IF(F1.GE.F)GO TO 6999
    IRES=0
    GO TO 6998
6999 IRES=1
6998 IF(F2.GE.F)GO TO 6997
    IRTS=0
    GO TO 6996
6997 IRTS=1
6996 K=IRTS+IRES
    IF(IRTS.EQ.0)GO TO 6880
    IF(K.EQ.0)IRO5=3
    IF(K.EQ.2)IRO5=1
    IF(K.EQ.1)IRO5=2

```



```

      IRES=IROS
      GO TO 6881
6880 IRES=3
6881 RETURN
      END
$IBFTC SY0142 DECK
      SUBROUTINE MPLOTS(NENTRY,NFNS,X,Y,NPNTS,NPRINT)
      INTEGER PRTER,DEVICE,CHAR(110),AX,AY,XCONST,YCONST,YAX
      INTEGER DUM(50)
      LOGICAL ERROR,GLINES,XAX,YAXON
      DIMENSION X(150,1),Y(150,1),NPNTS(10)
      PRTER=6
      IF(NENTRY.NE.0) GO TO 1
      ENTERR=3.1415927
      ERROR=.FALSE.
C      CALL MPLOTS(0,IGRID,XSIZE,YSIZE,NSBVL,NSBHL)
C      INPUT VARIABLES FROM FIRST ENTRY POINT AND CONVERT IF NECESSARY
      IXSIZE=X(1,1)
      IYSIZE=Y(1,1)
      NSBVL=NPNTS(1)
      NSBHL=NPRINT
      IGRID=NFNS
      IF(IXSIZE.GT.110) IXSIZE=110
      IF(IGRID.NE.2) GO TO 2
      IF((NSBVL.GT.0).AND.(NSBHL.GT.0)) GO TO 2
      B WRITE(PRTER,3)
      6 ERROR=.TRUE.
      RETURN
      2 IF((IXSIZE.GT.0).AND.(IYSIZE.GT.0)) GO TO 4
      WRITE(PRTER,5)
      GO TO 6
      4 IF((IGRID.GT.2).OR.(IGRID.LT.0)) GO TO 8
      RETURN
C      BEGINNING OF ENTRY POINT 1
C      CALL MPLOTS(1,NFNS,X(I,J),Y(I,J),NPNTS(K),NPRINT)
      1 IF(ENTERR.EQ.3.1415927) GO TO 200
C      IF ENTRY=0 NOT USED, USE EQUIVALENT OF
C      CALL MPLOTS(0,2,100,60,10,6)
      209 IXSIZE=100
      IYSIZE=60
      IGRID=2
      NSBVL=10
      NSBHL=6
      ENTERR=3.1415927
      200 IF(.NOT.ERROR) GO TO 207
      ERROR=.FALSE.
      GO TO 209
C      CALCULATE MAX AND MIN OF X AND Y ARRAYS
      207 XMIN=X(1,1)
      XMAX=XMIN
      YMIN=Y(1,1)
      YMAX=YMIN
      DO 204 I=1,NFNS
      NF=NPNTS(I)
      DO 204 J=1,NF

```

```

XJI=X(J,I)
YJI=Y(J,I)
XMIN=AMIN1(XMIN,XJI)
XMAX=AMAX1(XMAX,XJI)
YMIN=AMIN1(YMIN,YJI)
204 YMAX=AMAX1(YMAX,YJI)
    IF((ABS(XMAX - XMIN).GE.1.E-34) .AND. (ABS(YMAX - YMIN).GE.1.E-34)
1) GO TO 205
    WRITE(PRTER,208)
    RETURN
C   SET UP X AND Y SCALE FACTORS
205 XI=IXSIZE-1
    XSCALE=XI/(XMAX-XMIN)
    YI=IYSIZE-1
    YSCALE=YI/(YMAX-YMIN)
    IF(IGRID.EQ.0) GO TO 210
    AY=-99999
    AX=AY
    IF((XMAX.GE.0.) .AND. (XMIN.LE.0.)) AX=1.5-XSCALE*XMIN
    YAXON=.FALSE.
    IF((YMAX.LT.0.) .OR. (YMIN.GT.0.)) GO TO 370
    YAXON=.TRUE.
    AY=1.5-YSCALE*YMIN
370 GLINES=IGRID.EQ.2
    IF((AX).EQ.(-99999)) AX=1
    IF((AY).EQ.(-99999)) AY=1
C   XCONST IS FIRST (LOWEST) LINE X=CONST TO BE PLACED IN GRID
    XCONST=MOD(AX,NSBVL)
C   DITTO FOR YCONST
    YCONST=MOD(AY,NSBHL)
210 SCIN=1./YSCALE
    DELTA=.5*SCIN
C   BEGINNING OF ACTUAL GRAPH OUTPUT
    LINE=-1
212 LINE=LINE+1
    XI=LINE
    YP=YMAX-XI*SCIN
    YI=YP+DELTA
    XI=YP-DELTA
    XAX=.FALSE.
    IF(IGRID.EQ.0) GO TO 213
    IF((YI.GE.0.) .AND. (XI.LT.0.)) XAX=.TRUE.
    IF(.NOT. GLINES) GO TO 213
    LL=IYSIZE-LINE
    IF(MOD(LL,NSBHL).EQ.YCONST) XAX=.TRUE.
213 NC=-17997958192
    IF(XAX) NC=-818089008
    DO 214 I=1,IXSIZE
214 CHAR(I)=NC
    NC=27661634608
    IF(XAX) NC=17997958192
    IF(GLINES) GO TO 230
    IF((IGRID.EQ.1) .AND. YAXON) CHAR(AX)=NC
    GO TO 231
230 DO 215 I=XCONST,IXSIZE,NSBVL

```



```

215 CHAR(I)=NC
C***** PICK UP VALUES WHICH FIT ON THIS LINE
231 DO 216 I=1,NFNS
    K=NPNTS(I)
    NC=818089008+(I-1)*1073741824
    IF(K.EQ.0) GO TO 216
    DO 217 J=1,K
        YJI=Y(J,I)
        IF(YI.LT.YJI) GO TO 217
        IF(XI.GE.YJI) GO TO 217
        NP=(X(J,I)-XMIN)*XSCALE+1.5
        CHAR(NP)=NC
217 CONTINUE
216 CONTINUE
    IF(IGRID.EQ.0) GO TO 902
218 IF((YI.GE.0.) .AND. (XI.LT.0.)) GO TO 22/
    IF(XAX) GO TO 902
    WRITE(NPRINT,225)(CHAR(I),I=1,IXSIZE)
    GO TO 220
902 WRITE(NPRINT,222)YP,(CHAR(I),I=1,IXSIZE)
    GO TO 220
221 NC=-25514150960
    WRITE(NPRINT,223) NC,(CHAR(I),I=1,IXSIZE)
220 IF(LINE.NE.(IYSIZE-1)) GO TO 212
    DO 372 I=1,IXSIZE
372 CHAR(I)=-17997958192
    IF(GLINES) GO TO 271
    IF((IGRID.NE.1) .OR. (.NOT.YAXON)) GO TO 371
    CHAR(AX)=-26587892784
    WRITE(NPRINT,219) (CHAR(I),I=1,IXSIZE)
371 WRITE(NPRINT,272)
    RETURN
271 CONTINUE
    J=0
    DO 300 I=XCONST,IXSIZE,NSBVL
        J=J+1
        K=J
        IF(K.LE.10) GO TO 300
        LL=MOD(K,10)
        K=J-LL
        CHAR(I-1)=818089008+K*1073741824
        K=LL
300 CHAR(I)=818089008+K*1073741824
    WRITE(NPRINT,219)(CHAR(I),I=1,IXSIZE)
    J=0
    DO 393 I=XCONST,IXSIZE,NSBVL
        J=J+1
        IF(I.EQ.AX) GO TO 301
        XI=I
        XVAL=XMIN+XI/XSCALE
        WRITE(NPRINT,302) J,XVAL
        GO TO 393
301 WRITE(NPRINT,303) J
393 CONTINUE
    WRITE(NPRINT,272)

```

```

      RETURN
3  FORMAT(38H-'MPLOTS' ERROR IN GRID SPECIFICATIONS  //)
5  FORMAT(31H-'MPLOTS' A GRAPH DIMENSION = 0 //)
201 FORMAT(21H-'MPLOTS' ENTRY ERROR //)
208 FORMAT(43H-'MPLOTS' - ILLEGAL PLOT -- LIPE X=K OR Y=K //)
219 FORMAT(11X,110(A1))
222 FORMAT(1XE10.3,110A1)
223 FORMAT(5XA6,110(A1))
225 FORMAT(11X,110A1)
272 FORMAT(1H1)
302  FORMAT(5X12.3H = E10.3)
303  FORMAT(3X12.14H IS THE Y-AXIS  )
      END

```

APPENDIX C

Output Listing for Sample Data

Age Class Number 2 Extends from 25.0 to 35.0

This Tree Does Not Fit the Normal Population Defined
by this Class.

Age is 26.0, Height is 35.0, Test Statistic is -2.65

Tree Dropped from Calculation

The Following 94 Trees are Included in this Class

Tree Age	Tree Height
26.0	38.0
26.0	43.0
26.0	50.0
26.0	58.0
26.0	65.0
26.0	69.0
26.0	78.0
27.0	62.0
27.0	64.0
27.0	67.0
27.0	75.0
28.0	47.0
28.0	52.0
28.0	56.0
28.0	57.0
28.0	59.0
28.0	59.0
28.0	63.0
28.0	67.0
28.0	71.0
28.0	71.0
28.0	75.0
29.0	48.0
29.0	50.0
29.0	50.0
29.0	53.0
29.0	56.0
29.0	57.0
29.0	58.0
29.0	60.0
29.0	63.0
29.0	65.0
29.0	65.0

Tree Age		Tree Height		
29.0		67.0		
29.0		71.0		
29.0		74.0		
29.0		76.0		
29.0		80.0		
30.0		52.0		
30.0		53.0		
30.0		54.0		
30.0		55.0		
30.0		56.0		
30.0		65.0		
30.0		65.0		
30.0		73.0		
30.0		79.0		
30.0		82.0		
30.0		82.0		
30.0		85.0		
31.0		55.0		
31.0		47.0		
31.0		53.0		
31.0		54.0		
31.0		58.0		
31.0		59.0		
31.0		60.0		
31.0		60.0		
31.0		61.0		
31.0		61.0		
31.0		61.0		
31.0		62.0		
31.0		63.0		
31.0		64.0		
31.0		66.0		
31.0		67.0		
31.0		76.0		
31.0		78.0		
31.0		82.0		
32.0		48.0		
32.0		51.0		
32.0		56.0		
Sum Ages	Sum Age Sqd	Mean	Stdev	Coeff Var
8037.0	412373.0	51.2	2.4681	14.85
Sum of Ht	Sum of Htsq	St Dev Ht		
12658.0	1042912.0	11.9751		

Age Class Number 5 Extends from 55.0 to 65.0
The Following 47 Trees are Included in this Class

Tree Age	Tree Height
56.0	60.0
56.0	63.0
56.0	68.0
56.0	74.0
56.0	76.0
56.0	77.0
56.0	82.0
56.0	91.0
56.0	92.0
56.0	95.0
56.0	96.0
57.0	70.0
57.0	73.0
57.0	80.0
57.0	82.0
57.0	83.0
57.0	85.0
57.0	89.0
57.0	96.0
57.0	99.0
57.0	102.0
57.0	104.0
58.0	65.0
58.0	67.0
58.0	76.0
58.0	77.0
58.0	79.0
58.0	84.0
58.0	85.0
58.0	87.0
58.0	88.0
58.0	94.0
58.0	97.0
58.0	102.0
58.0	103.0
59.0	73.0
59.0	77.0
59.0	85.0
59.0	87.0
60.0	80.0
61.0	65.0
61.0	74.0
62.0	87.0
63.0	82.0
63.0	97.0
63.0	102.0
63.0	104.0

Sum Ages	Sum Age Sqd	Mean	Stdev	Coeff Var
2729.0	158653.0	58.1	2.0684	14.30
	Sum of Ht	Sum of Htsq	St Dev Ht	
	3954.0	339296.0	12.0283	

Age Class Number 6 Extends from 65.0 to 75.0
The Following 6 Trees are Included in this Class

Tree Age		Tree Height		
	67.0		85.0	
	68.0		93.0	
	68.0		114.0	
	69.0		64.0	
	69.0		95.0	
	73.0		109.0	
Sum Ages	Sum Age Sqd	Mean	Stdev	Coeff Var
414.0	28588.0	69.0	2.0976	19.20
	Sum of Ht	Sum of Htsq	St Dev Ht	
	560.0	53872.0	17.9183	

Age Class Number 7 Extends from 75.0 to 85.0
The Following 5 Trees are Included in this Class

Tree Age		Tree Height		
	77.0		78.0	
	77.0		90.0	
	78.0		92.0	
	79.0		105.0	
	82.0		96.0	
Sum Ages	Sum Age Sqd	Mean	Stdev	Coeff Var
393.0	30907.0	78.6	2.0737	10.64
	Sum of Ht	Sum of Htsq	St Dev Ht	
	461.0	42889.0	9.8082	

The Following Tree(s) too Low for Any Age Class,
Dropped from Calculations

Age
4.0
5.0
6.0
7.0
13.0
14.0
15.0

The Following Tree(s) too High for Any Age Class,
Dropped from Calculations

Age
108.0
88.0

Normal Forward Solution Regression Matrix
Regression Equation in the Form $-CV=A+BX$, $X=Age^{**}-\frac{1}{2}$
Coefficients are, $A = -2.43415$ $B = 128.96366$

	A	X	Y
A	7.0	1.0564	119.1932
X		0.1674	19.0120
Y			2256.5054
X		0.0079	1.0247
Y			226.9307
RES			94.7876

Normal Forward Solution Regression Matrix
Regression Equation in the Form $-CV=A+BX$, $X=Age$ Squared
Coefficients are, $A = 21.85386$ $B = -0.00169$

	A	X	Y
A	7.0	20022.7087	119.1932
X		83044065.0000	297455.3047
Y			2256.5054
X		25771370.5000	-43483.4180
Y			226.9307
RES			153.5622

Normal Forward Solution Regression Matrix
Regression Equation in the Form $-CV=A+BX$, $X=Age$
Coefficients are, $A = 26.67247$ $B = -0.19285$

	A	X	Y
A	7.0	350.0894	119.1932
X		20022.7087	5476.4085
Y			2256.5054
X		2513.7654	-484.7750
Y			226.9307
RES			133.4427

Normal Forward Solution Regression Matrix
 Regression Equation in the Form— $CV=A+BX$, $X=Age^{**1/2}$
 Coefficients are, $A = 11.87496$ $B = 7429.29553$

	A	X	Y
A	7.0	0.0049	119.1932
X		0.0000	0.1048
Y			2256.5054
X		0.0000	0.0222
Y			226.9307
RES			62.1702

Anov.					
Source	Df	S of S	Mean Sqs	Fc	Ft
Mean	1.	2030.	2030.	163.2274	6.6100
Ind Var	1.	165.	165.	13.2508	6.6100
Residual	5.	62.	12.		
Total	7.	2257.			

Fter Regression and Anov, the Following Equation Remains
 Valid — $CV = 11.8750 + 7429.2955X$, X A Function of Age

	A	X	Y
A	384.0	9.30494	1651.19249
X		0.24865	39.57960
Y			7117.52087
X		0.02318	-0.43147
Y			17.42554
RES			9.39431

In the Matrix Above, $X = Age$, $Y = Height$, $A = Number$
 of Observations
 Guide Curve of Total Height Over Age — $Ht =$

Guide Curve of Total Height Over Age.

Prunus Serotina--Black Cherry Data from Northern West Virginia-H.H. Heesch

1 is the Y-Axis
 2 = 0.266E 02
 3 = 0.325E 02
 4 = 0.385E 02
 5 = 0.444E 02
 6 = 0.504E 02
 7 = 0.564E 02
 8 = 0.623E 02
 9 = 0.683E 02
 10 = 0.742E 02

Prunus Serotina--Black Cherry Data from Northern West Virginia-H. H. Heesch

```
1 is the Y-Axis
2 = 0.564E 02
```

Site Index Curves For,
Prunus Serotina--Black Cherry Data from Northern West Virginia-H. H. Heesch

1 is the Y-Axis
 2 = 0.244E 02
 3 = 0.285E 02
 4 = 0.325E 02
 5 = 0.366E 02
 6 = 0.406E 02
 7 = 0.446E 02
 8 = 0.487E 02
 9 = 0.527E 02
 10 = 0.568E 02

The Following Arrays were Used to Plot the Site Index Curves
Site Index

Age	50.0	60.0	70.0	80.0	90.0	100.0	110.0	120.0
20.0	10.7	22.5	34.2	45.9	57.7	69.4	81.1	92.9
21.0	13.3	24.9	36.4	48.0	59.6	71.1	82.7	94.3
22.0	15.7	27.1	38.5	49.9	61.4	72.8	84.2	95.6
23.0	18.0	29.2	40.5	51.8	63.1	74.4	85.6	96.9
24.0	20.1	31.3	42.4	53.6	64.7	75.9	87.0	98.2
25.0	22.2	33.2	44.2	55.2	66.3	77.3	88.3	99.4
26.0	24.1	35.0	45.9	56.8	67.8	78.7	89.6	100.5
27.0	25.9	36.7	47.5	58.4	69.2	80.0	90.8	101.6
28.0	27.6	38.3	49.1	59.8	70.5	81.3	92.0	102.7
29.0	29.2	39.9	50.5	61.2	71.8	82.5	93.1	103.8
30.0	30.8	41.3	51.9	62.5	73.1	83.6	94.2	104.8
31.0	32.2	42.7	53.2	63.7	74.3	84.8	95.3	105.8
32.0	33.6	44.0	54.5	64.9	75.4	85.8	96.3	106.7
33.0	34.9	45.3	55.7	66.1	76.5	86.9	97.3	107.7
34.0	36.2	46.5	56.8	67.2	77.5	87.9	98.2	108.6
35.0	37.3	47.6	57.9	68.2	78.6	88.9	99.2	109.5
36.0	38.5	48.7	59.0	69.3	79.5	89.8	100.0	110.3
37.0	39.6	49.8	60.0	70.2	80.5	90.7	100.9	111.1
38.0	40.6	50.8	61.0	71.2	81.4	91.5	101.7	111.9
39.0	41.6	51.7	61.9	72.1	82.2	92.4	102.5	112.7
40.0	42.5	52.6	62.8	72.9	83.1	93.2	103.3	113.5
41.0	43.4	53.5	63.6	73.7	83.9	94.0	104.1	114.2
42.0	44.3	54.4	64.5	74.5	84.6	94.7	104.8	114.9
43.0	45.1	55.2	65.2	75.3	85.4	95.5	105.5	115.6
44.0	45.9	55.9	66.0	76.1	86.1	96.2	106.2	116.3
45.0	46.6	56.7	66.7	76.8	86.8	96.9	106.9	117.0
46.0	47.4	57.4	67.4	77.5	87.5	97.5	107.6	117.6
47.0	48.1	58.1	68.1	78.1	88.2	98.2	108.2	118.2
48.0	48.7	58.7	68.8	78.8	88.8	98.8	108.8	118.8
49.0	49.4	59.4	69.4	79.4	89.4	99.4	109.4	119.4
50.0	50.0	60.0	70.0	80.0	90.0	100.0	110.0	120.0
51.0	50.6	60.6	70.6	80.6	90.6	100.6	110.6	120.6
52.0	51.2	61.2	71.2	81.1	91.1	101.1	111.1	121.1
53.0	51.7	61.7	71.7	81.7	91.7	101.7	111.7	121.7
54.0	52.3	62.3	72.2	82.2	92.2	102.2	112.2	122.2
55.0	52.8	62.8	72.8	82.7	92.7	102.7	112.7	122.7
56.0	53.3	63.3	73.3	83.2	93.2	103.2	113.2	123.2
57.0	53.8	63.8	73.7	83.7	93.7	103.7	113.7	123.7
58.0	54.2	64.2	74.2	84.2	94.2	104.2	114.2	124.1
59.0	54.7	64.7	74.7	84.7	94.6	104.6	114.6	124.6
60.0	55.1	65.1	75.1	85.1	95.1	105.1	115.1	125.1

APPENDIX D

Definition of Variables

Although SICURV has been tested and seems to function as expected, it may not be able to handle all situations. For this reason, a definition of variables is given which will facilitate any changes a programmer wishes to make.

I MAIN PROGRAM — SICURV

INCNT: The number of paired observations of height and age (US)*.

ALB: The lower limit of the youngest age class (US).

AGINT: The age interval of each age class (US).

AUB: The upper bound of the oldest age class (US).

NFNS: The number of site index curves to be plotted (US).

SIMIN: The index of the lowest site index curve (US).

SIINT: The interval between site index curves (US).

SIMAX: The index of the highest site index curve (US).

SILOW: The lowest age over which the site index curves are to be plotted (US).

SIHIGH: The oldest age over which the site index curves are to be plotted (US).

F: The tabular value for the F-test with $2, N-1$ degrees of freedom; N = Number of age classes (US).

ANM(LO): An alphanumeric array used to label output (US).

RNB(I): Format specifications for the paired observations (US).

IRC: Denotes the order of input on each card. If $IRC = 1$, Height precedes age.

HT(I): The height of the I^{th} tree in the input data.

AG(I): The age of the I^{th} tree in the input data.

SORT: A variable which represents the upper bound of the KK^{th} age class during the sorting procedure.

KK: Counts the total number of age classes into which the data is sorted.

MM: Counts the number of trees too young for any age class.

JL: Counts the number of trees too old for any age class.

NOGO: Program Control value. If an age class has been dropped from calculations, this value will terminate processing at the end of the sorting procedure.

ALT: Represents the lower bound of the KK^{th} age class during sorting.

*(US) denotes user specifies.

SOHT: The sum of the heights of all trees in an age class.

SOHTSQ: The sum of squares of the heights of all trees in an age class.

JJ: Counts the number of trees rejected in an age class during Z-tests on that class.

P: Counts the number of trees in an age class.

L: Counts the number of trees in an age class.

SUMAGE: The sum of the ages of all trees in an age class.

SAGES: The sum of squares of all trees in an age class.

IAG(L): Represents the I^{th} value of the L^{th} tree in an age class.

JK: Represents the JK^{th} tree in an age class. This value is used only when the number of trees in an age class is less than 3.

STDEV2: The standard deviation of heights of all trees in an age class.

HTMEAN: Represents the mean heights of all trees in an age class.

I: An integer counter used for various tasks throughout the program.

IT: The I^{th} value of the L^{th} tree in the Z-test.

ZTEST: The calculated Z value for each tree.

IMIN: The number of trees remaining in any age class after rejections by the Z-test.

IM: The L^{th} number of the I^{th} tree remaining in an age class after rejections by the Z-test. IM is used only if less than three trees remain in an age class after rejections by the Z-test.

AMEAN: The average age of all trees remaining in an age class.

G(KK): An array of mean ages for each age class.

STDEV: The standard deviation of all ages in an age class.

INBR: The number of trees in an age class after all rejections have been made.

NN: Represents the I^{th} tree of the input array in relation to the L^{th} tree of the KK^{th} age class.

TOHIGH (JL): The age of the JL^{th} tree too old for any age class.

TOLOW (MM): The age of the MM^{th} tree too young for any age class.

ALCL: The midpoint of the KK^{th} age class.

CV(KK): The coefficient of variation of height for the KK^{th} age class.

INBR: The number of trees remaining in an age class after rejections by the Z-test.

AS (KK): The square of the mean age of the KK^{th} age class.

- D(KK): The reciprocal of the square root of the mean age of the KK^{th} age class.
- CV(KK): The coefficient of variation of the KK^{th} age class.
- SUCVS: The sum of all CV(KK).
- SUAS: The sum of all AS(KK).
- SUMAG: The sum of all G(KK).
- SURAS: The sum of all D(KK).
- CVS: The sum of all CV(KK)².
- SGS: The sum of all G(KK)².
- SASAS: The sum of all AS(KK)².
- SRARA: The sum of all D(KK)².
- CVX: The sum of all CV(KK) * G(KK).
- DVXS: The sum of all CV(KK) * AS(KK).
- CVRXS: The sum of all CV(KK) * D(KK).
- ICURV: A value used for printout control. It represents the type of equation on which the regression is being performed for Coefficient of Variation.
- B(KK): The reciprocal of the square of mean age of the KK^{th} age class.
- CVB: The sum of all CV(KK) * B(KK).
- BSUM: The sum of all B(KK).
- BS: The sum of all B(KK)².
- RES1: The residual sum of squares for the regression of Coefficient of Variation when $X = \text{age}$.
- RES2: As above except when $X = \text{age}^2$.
- RES3: As above except when $X = 1/\sqrt{\text{age}}$.
- RES4: As above except when $X = 1/\text{age}^2$.
- GDCRV: The smallest of the above residuals.
- C1: The y intercept of the equation for Coefficient of Variation for any transformation of age.
- C2: The slope of the above equation for any transformation of age.
- A1: The equivalent of C1 above if X in the regression for Coefficient of Variation is mean age.
- A2: The equivalent of C2 above when X in the regression equation for Coefficient of Variation is mean age.
- A3: As for A1, but when $X = \text{age}^2$.
- A4: As for A2, but when $X = \text{age}^2$.
- A5: As for A1, but when $X = 1/\sqrt{\text{age}}$.
- A6: As for A2, but when $X = 1/\sqrt{\text{age}}$.
- A7: As for A1, but when $X = 1/\text{age}^2$.
- A8: As for A2, but when $X = 1/\text{age}^2$.

IRES: Represents the significance detected by the subroutine VARAN, on the coefficients of the equation $CV = a + bx$.
PLCV(J,1): An array of values which represents the regression of Coefficient of Variation. This array is plotted by the MPLOTS subroutine.
NSBVL: Specifies the type of background used by the MPLOTS subroutine.
PLAC(I,J): An array of ages over which PLCV(I,1), PLGC(I,1) will be plotted.
NPNTS(J): The number of points to be plotted on the J^{th} graph.
CV50: The value of PLCV(I,1) when age is 50 years.
IO: The value of I when PLAC(I,J) is equal to SILOW.
NPCV: The number of functions to be plotted by MPLOTS for CV. A constant value of 1.
J: Counts the number of trees not rejected during the sorting procedure.
PP: Similar to J above.
RAGE(J): The J^{th} reciprocal of all AG(I) which were not previously rejected.
ELGHT(J): The J^{th} natural logarithm of all Ht(I) not previously rejected.
SRAGE: The sum of all RAGE(J).
SLHT: The sum of all ELGHT(J).
SRAGES: The sum of all $RAGE(J)^2$.
SLHTS: The sum of all $ELGHT(J)^2$.
SRALHT: The sum of all $ELGHT(J) * RAGE(J)$.
SRASRA: The sum of the RAGE(J) with the effects of the mean as a source of variation removed.
SAXHRA: The sum of the $RAGE(J) * ELGHT(J)$ with the effects of the mean as a source of variation removed.
SHTSRA: The sum of the $ELGHT(J)^2$ with the effect of the mean as a source of variation removed.
GCRES: The residual sum of squares for the guide curve.
GC1: The y intercept of the regression equation $Ht = e^{(a + b/AGE)}$.
GC2: The slope of the regression equation $Ht = e^{(a + b/AGE)}$.
PLGC(I,1): The height of the guide curve at the I^{th} age.
GCHT: The height of the guide curve at age 50.
JO: The value of I when PLAC(I,L) is equal to SILOW.
NONS: The number of guide curves to be plotted. It has a constant value of 1.
PLSI(KK,JJ): The KK^{th} height of the JJ^{th} site index curve to be plotted.

PLOC(KK,JJ): The KK^{th} age over which the JJ^{th} site index curve will be plotted.

ACON: A constant value for each site index curve equal to $((\text{SIMIN}/\text{GCHT}) - 1) / \text{CV50}$. SIMIN is incremented by the value of SIINT as each curve is calculated.

SIHOLD: This holds the original value of SIMIN for printout control.

IHOLD: Similar to above but for the counter JO.

NP: The number of points to be plotted in each site index curve.

NPNTS(JJ): The number of points in the JJ^{th} site index curve, always equal to NP.

H(I): The index of the I^{th} site index curve. A maximum of 10 values will be stored in this array.

II-A FUNCTION SUBPROGRAM SOMS

This function Subprogram sums the real values passed to it. The arguments are:

Z: A real array whose sum is desired.

NL: The number of values in array Z.

Computes and returns

$$\begin{aligned} & \text{NL} \\ \text{SOMS} &= \sum_{I=1} Z(I) \end{aligned}$$

II-B FUNCTION SUBPROGRAM SOS

SOS Computes the sum of squares of the arguments passed to it. The arguments are:

A: An array of real values whose sum of squares is desired.

KK: The number of values in array A.

Computes and returns

$$\begin{aligned} & \text{KK} \\ \text{SOS} &= \sum_{I=1} A(I)^2 \end{aligned}$$

II-C FUNCTION SUBPROGRAM SCP

This function subprogram computes the sum of the cross-products of the real arguments passed to it. The arguments are:

FA: An array of real values.

FB: An array of real values.

IJ: The number of values in each array.

Computes and returns

$$\text{SCP} = \sum_{I=1}^{IJ} \text{FA}(I) * \text{FB}(I)$$

III-A SUBROUTINE SUBPROGRAM LIREG

This subroutine calculates a normal forward solution regression matrix for the equation $CV = a + bx$. The values passed to the program are:

RP: The number of observations.

XS: The sum of the observations of the independent variable.

YS: As above for the dependent variable.

XSS: The sum of squares of the independent variable.

YSS: As above for the dependent variable.

XSYS: The sum of the cross-products of the independent and dependent variable.

ICURV: Previously defined (Main Program).

The values calculated in the subroutine are:

SXSRA: The sum of squares of the independent variable with the effect of the mean removed.

SYSRA: As above but for the independent variable.

SXYRA: As above but for the cross-product of the two variables.

CURES1: The residual sum of squares of the regression matrix.

CV1: The y intercept of the equation $CV = a + bx$.

CV2: The slope of the above equation.

The values CV1, CV2, and CVRES1 are returned to the main program.

III-B SUBROUTINE SUBPROGRAM VARAN

Varan performs an analysis of variance on the equation $CV = a + bx$ to determine the significance of the regression coefficients. The arguments passed to the subroutine are:

V: The number of observations on the independent variable.

V1: The sum of squares of the dependent variable.

- V2: The sum of cross-products of the dependent and independent variable.
- V3: The sum of the observations on the independent variable.
- V4: As above, but for the dependent variable.
- F: The F value read in at the beginning of the program.
- CAT: The slope of the regression equation.

The values calculate in the subroutine are:

- ASS: The sum of squares of the mean.
- VARSS: The sum of squares of the independent variable.
- RSS: The residual sum of squares.
- RMS: The residual mean squares.
- RDF: The residual degrees of freedom.
- F1: The variance ratio between the ASS and RMS.
- F2: The variance ratio between VARSS and RMS.
- IRES: An integer which indicates the significance, partial significance or non-significance of the regression coefficients. This value is returned to the main program.

III-B SUBROUTINE SUBPROGRAM MPLOTS

This subroutine subprogram was written by Jim Mitchell at the University of Waterloo. MPLOTS plots the graphs of coefficient of variation, the guide curve, and the site index curves. No attempt will be made to define the variables used in MPLOTS. All calling arguments to the subroutine are specified by the user or calculated by the main program. The MPLOTS SUBROUTINE will plot up to a maximum of 10 functions with up to 110 points per function.

IV INTERNAL FUNCTION SUBPROGRAMS

SICURV uses IBM library function subprograms in various parts of the program. Their names and purposes are:

- AMIN1: Picks and returns the minimum value of all arguments passed to it.
- AMAX1: As above for the maximum value.
- ALOG: Computes and returns the natural logarithm of the argument passed.
- MOD: Computes and returns the modulus (integral part) of the two arguments passed.
- ABS: Computes and returns the absolute value of the argument passed.

EXP: Raise the base of the naperian logarithm (e) to the power of the argument passed and returns the result.

SICURV also uses **FPTTRA**, a **WVUCC** library subprogram, to allow unlimited floating point traps. This is set in the program to convert all values to Zero if they are near Zero, to suppress all error message and to return to execution. **FPTTRA** was written by H. L. Pearson.

Appendix E. MACRO FLOW CHART FOR SICURV

APPENDIX F. Literature Cited

1. Bruce, Donald, and F. X. Schumacher
1950. Forest Mensuration. Third Edition, McGraw-Hill Book Co., 483 pp., illus.
2. Kulow, D. L., D. W. Sowers, and H. H. Heesch
1966. Site Index Curves for Virginia Pine in West Virginia. W.V.U. Agr. Exp. Sta. Bulletin 536T.