287
4-28-82
JMK

I-2775

(2)

ornl

ORNL-5813

OAK
RIDGE
NATIONAL
LABORATORY

UNION
CARBIDE

DR. 454

MASTER

An Interactive Statistical
Distribution Analysis
Program Utilizing Numerical
and Graphical Methods

S. R. Glandon
D. E. Fields

DO NOT MICROFILM
COVER

OPERATED BY
UNION CARBIDE CORPORATION
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road, Springfield, Virginia 22161
NTIS price codes—Printed Copy: A05 Microfiche A01

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ORNL--5813

DE82 013341

ORNL-5813

Contract No. W-7405-eng-26

HEALTH AND SAFETY RESEARCH DIVISION

AN INTERACTIVE STATISTICAL-DISTRIBUTION-
ANALYSIS PROGRAM UTILIZING NUMERICAL
AND GRAPHICAL METHODS*

S. R. Glandon
D. E. Fields

Date Published: April 1982

DISCLAIMER

This book was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Prepared for the
Office of Reactor Research and Technology
Fast Reactor Safety Program,
U. S. Department of Energy

Prepared by the
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37830
operated by
UNION CARBIDE CORPORATION
for the
DEPARTMENT OF ENERGY

EB
DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

CONTENTS

	<u>Page</u>
LIST OF FIGURES.	iv
LIST OF TABLES	iv
ACKNOWLEDGEMENTS	v
ABSTRACT.	vi
1. INTRODUCTION	1
2. STATISTICAL METHOD USED IN TERPED/P.	2
3. GRAPHICS SUPPORTED IN TERPED/P	7
4. DISCUSSION OF CODE USAGE	9
5. TERPED/P PROGRAM STRUCTURE	12
REFERENCES	16
APPENDIX A. Sample Run.	17
APPENDIX B. Sample DISSPLA Plot Summary	27
APPENDIX C. Preview of Compressed Plot Data File.	33
APPENDIX D. High-Resolution Plot produced by TERPED/P	39
APPENDIX E. FORTRAN-10 Listing of TERPED/P Code	45
APPENDIX F. Input Data Set for Sample Run	77
APPENDIX G. Modifications for Systems Without DISSPLA Graphic Software	81

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1	TERPED/P subprogram structure.	13

LIST OF TABLES

<u>Table</u>		<u>Page</u>
1	Some useful statistical parameters for lognormally distributed data (Aitchison and Brown, 1969).	4
2	List of error messages generated by TERPED/P.	9

ACKNOWLEDGEMENTS

The authors wish to thank Robin Smith, the typist who deciphered our handwriting, and Wilma Minor, who prepared the final manuscript. Also, we acknowledge with appreciation Fred Baes III for his advice on log-scaled printer plots and review of this manuscript, and C. W. Miller for his review of the manuscript. Special thanks are extended to George L. Sherwood, Office of Nuclear Programs, U. S. Department of Energy, for his support.

ABSTRACT

The TERPED/P program is designed to facilitate the quantitative analysis of experimental data, determine the distribution function that best describes the data, and provide graphical representations of the data. This code differs from its predecessors, TEDPED and TERPED, in that a printer-plotter has been added for graphical output flexibility. The addition of the printer-plotter provides TERPED/P with a method of generating graphs that is not dependent on DISSPLA, Integrated Software Systems Corporation's confidential proprietary graphics package. This makes it possible to use TERPED/P on systems not equipped with DISSPLA. In addition, the printer plot is usually produced more rapidly than a high-resolution plot can be generated.

Graphical and numerical tests are performed on the data in accordance with the user's assumption of normality or lognormality. Statistical analysis options include computation of the chi-squared statistic and its significance level and the Kolmogorov-Smirnov one-sample test "confidence level" for data sets of more than 80 points. Plots can be produced on a Calcomp paper plotter, a FR80 film plotter, or a graphics terminal using the high-resolution, DISSPLA-dependent plotter or on a character-type output device by the printer-plotter. The plots are of cumulative probability (abscissa) versus user-defined units (ordinate).

The program was developed on a Digital Equipment Corporation (DEC) PDP-10 and consists of 1500 statements. The language used is FORTRAN-10, DEC's extended version of FORTRAN-IV.

1. INTRODUCTION

TERPED/P is a version of the TERPED interactive statistical analysis and plotting program (Fields, 1981) that has been modified to include printer-plotter output of graphical data. TERPED/P and its predecessors, TERPED and TEDPED (Fields, Little, and Shaeffer, 1980), were designed to assist in determining whether data were distributed normally or log-normally by performing various numerical and graphical tests. For example, this information could be useful for the evaluation of environmental standards, the assessment of environmental impacts, and the interpretation of the output of mathematical models (Fields, 1981).

All three programs are interactive and are designed to be controlled from a local terminal. TEDPED requires display of the graphical information generated by the program on a graphics terminal such as the TEKTRONIX 4006. The TERPED program generates "plot data sets" through the use of subroutines provided by DISSPLA*. These plot data sets can be routed to any high resolution plotting device such as the FR80 film plotter and the Calcomp plotter, or can be viewed on a graphics terminal. TERPED also performs certain bookkeeping functions that are left to the TEDPED user.

TERPED/P, with its printer-plotter capability, is the result of an attempt to give the TEDPED/TERPED programs one form of graphical data output that is independent of the DISSPLA subroutines, to which many potential users may not have access.

TERPED/P provides the user with a numerical and graphical description of a set of data corresponding to an assumption of either normality or log-normality. The plots produced, both high resolution and printer-plotter, are of linearized cumulative probability with a fitted line showing how well a set of data fits the assumed distribution. Options include a Kolmogorov-Smirnov one-sample test and an algorithm for computing the chi-squared statistic. Data can be input either by hand or

*A proprietary software distributed by ISSCO (Integrated Software Systems Corporation).

from a disk file. The code is written in FORTRAN-10 for execution on a Digital Equipment Corporation PDP-10 computer. Typical central processor unit execution time is 0.32 seconds, exclusive of plotting time. The code size is approximately 1500 card images.

TERPED/P should prove useful even to SAS* users. Even though both offer printer-plots, SAS cannot produce high resolution plots and is intended for batch processing applications. TERPED/P was designed to run in a file-oriented, time-sharing computer environment.

2. STATISTICAL METHOD USED IN TERPED/P

Probably the simplest method of determining the distribution function of a set of data is to calculate the cumulative probability values and plot them on special graph paper with non-linearly scaled cumulative probability axis. A straight line means that the data are distributed according to the assumed distribution function. The ordinate is scaled according to the assumed distribution function; if the data are assumed normally distributed, the ordinate is linear; if lognormally distributed, the ordinate is logarithmic.

TERPED/P carries out this operation and provides other useful statistics as well. Statistics are computed and output to the user corresponding to an assumed distribution of either normality or log-normality, selected at the start of the program. This hypothesis controls all subsequent operations performed by the code.

The code first computes the cumulative probability (P_i) by ordering the set of m data points by increasing magnitude with the index i and using the following equation (Kimball, 1960):

$$P_i = (i - 0.375)/(m + 0.25). \quad (1)$$

*Proprietary software distributed by SAS Institute, Inc., of Raleigh, North Carolina.

The user's data are then transformed into a linear function of cumulative probability assuming the hypothesized normal or lognormal distribution. Data linearization is by a spline fit (Ahlberg, Nelson, and Walsh, 1967) of the computed probability of occurrence values to a set of tabulated inverse error function values representing the number of standard deviations away from the mean value (50% cumulative probability value).

If the user hypothesized that the data were distributed normally, the arithmetic mean ($\bar{x}$) and the variance about the mean (s^2) are computed according to the following equations:

$$\bar{x} = m^{-1} \sum_{i=1}^m x_i \quad (2)$$

and

$$s^2 = (m-1)^{-1} \sum_{i=1}^m (\bar{x} - x_i)^2, \quad (3)$$

where $x_i; i=1, m$ is a set of normal measured variables. If the assumption was that the data were lognormally distributed, then several relevant statistical parameters are computed. The equations are shown in Table 1. These include the mean of the logs ($\hat{\mu}$), the most probable value (x_p), the median value (x_m), the arithmetic mean value ($\bar{x}$), the 99th quantile value (x_{99}), and the log variance ($\hat{\sigma}^2$).

The calculated "expectation" value ($\langle x \rangle$) of an observable quantity is $\bar{x}$ or x_p if the distribution is normal or lognormal, respectively. The value of $\langle x \rangle$ would not be expected to be either $\bar{x}$ or x_p if the distribution was otherwise.

After the cumulative probability values are transformed into a linear distribution of standard deviations from mean values and other statistics computed and output, the values are fit to a straight line using a linear least-squares algorithm (Westley and Watts, 1970).

Table 1. Some useful statistical parameters for lognormally distributed data (Aitchison and Brown, 1969)

$$\hat{\mu} = m^{-1} \sum_{i=1}^m \ln x_i$$

$$\hat{\sigma}^2 = (m - 1)^{-1} \sum_{i=1}^m (\ln x_i - \hat{\mu})^2$$

$$x_p = \exp (\hat{\mu} - \hat{\sigma}^2)$$

$$x_m = \exp (\hat{\mu})$$

$$\bar{x} = \exp (\hat{\mu} + \hat{\sigma}^2/2)$$

$$x_{99} = \exp (\hat{\mu} + 2.326\hat{\sigma})$$

The agreement between the resulting fit and the data is indicated by computing Pearson's correlation coefficient. If $x_i: i=1, m$ are data ordinates with mean $\bar{x}$, and $y_i: i=1, m$ are least-squares fit ordinates with mean $\bar{y}$ at the abscissa of point i , then the correlation coefficient is defined as:

$$r = \frac{\sum_{i=1}^m (\bar{x} - x_i)(\bar{y} - y_i)}{\left[\sum_{i=1}^m (\bar{x} - x_i)^2 \sum_{i=1}^m (\bar{y} - y_i)^2 \right]^{1/2}} \quad (4)$$

by Snedecor and Cochran in *Statistical Methods*. Values of r will range from 0 for a very poor fit to 1.0 for a perfect fit.

There are two optional statistical tests performed by TERPED/P. The chi-squared goodness-of-fit test (Kendal and Stuart, 1961) and the Kolmogorov-Smirnov one-sample test analysis (Bradley, 1968) are two added means of assessing the validity of the hypothesis. If selected, these tests are performed before the linear least-squares fit and computation of Pearson's correlation coefficient.

The goodness of fit between data and the fitted curve having the hypothesized distribution may be calculated using a chi-squared test. The normalized data are represented as lying in n cells, where n may be chosen to be equal to or less than the number of data points. The "observed" value for each cell is the count s_i , the number of data points lying in the cell. This is compared to t_i , the expected number for a normal or lognormal distribution having the same mean and variance. The comparison is made by calculating the observed chi-squared statistic, defined as:

$$\chi^2 = \sum_{i=1}^n (s_i - t_i)^2 / t_i, \quad (5)$$

which is based on $(n-3)$ degrees of freedom. From this statistic the "significance level" for rejecting the "null hypothesis" that the two sets of numbers have the same distribution can be computed. The greater the likelihood that the data are distributed as assumed, the greater the significance level.

The significance level can be considered to be the probability of calculating, based on random fluctuations alone, a greater value of χ^2 than the one calculated here. Frequently, the 0.05 level of significance is used to make a decision about the assumed distribution. If the significance is greater than 0.05, the user can reasonably conclude that in using the chi-squared test he cannot show his data to be different from a normal or lognormal distribution (whichever assumed) with the same mean and variance (or log mean and log variance) as his data.

The Kolmogorov-Smirnov (KS) one-sample test considers the hypothesis of equality between the actual distribution of data and the assumed normal or lognormal distribution. The KS test is based on the maximum difference between corresponding terms of the value ranked sets x_i and y_i . We define $D = \max_{i=1,m} (x_i - y_i)$. This value is compared to an expected statistic valid for that number of points and a "confidence level" is determined. The KS confidence level returned by TERPED/P is the probability of Z not being exceeded if the hypothesis of equality is true and if the alternative is "two-sided," i.e., if a given measurement has an equal chance of lying above or below the expected value. Z is the quotient of D and the square root of the number of points ($\sqrt{n}$). The numerical procedure used in TERPED/P to compute the confidence level actually returns an asymptotic value which may be considered accurate for $m > 80$. A warning message is printed if the number of points is less than 80, and the user should consult tabulated confidence values.

In summary, TERPED/P consists of algorithms that compute several measures of central tendency, the chi-squared probability, the KS statistic, and the correlation coefficient.

3. GRAPHICS SUPPORTED IN TERPED/P

One of the most useful features of the TERPED/P program is its ability to generate plots of user-supplied data versus cumulative probability. These allow viewing of the data in collated form for easy recognition of general trends, outliers, and invalid data. Also, they provide references for comparison of old and new data. The abscissa is scaled in cumulative probability such that data will lie along a straight line if distributed as hypothesized. A least-squares fit of the linearized data is plotted on the same graph to permit visual evaluation of the distribution hypothesis.

There are two plot types supported in the TERPED/P program. The first is a printer-plot, which can be routed to any serial character-oriented output device. The second plot type is a high-resolution plot for either a paper or film plotter or a graphics terminal such as the Tektronix 4006.

There are three devices available for printer-plot output. The first is a line printer. When this device is chosen, the output of the printer-plot generator is routed into a disk file called a "spooled output file" with the name "*xxxxx.LPT" where x represents alphanumeric characters and * represents an alphabetic character. This file is automatically queued to the line printer when the user logs off the computer using the K/F command. Each plot is given a different file name, which is selected by the system.

The second device that can be accessed by the printer-plotter is the disk storage system. The file/device number is selected by the user. Under no circumstances should the data file and the output file be given same number as the output overwrites the data. These files are named "FORxx.DAT" where xx represents the file/device number selected by the user. This type of file is useful for long-term storage of printer plots.

The third device that can be accessed is the user's terminal. This selection allows immediate viewing of the plot without exiting the program. This feature can be useful as an aid to setting plot parameters

before generating a high-resolution plot or for simply looking at a distribution before testing another hypothesis against the data.

The printer-plot width is selected by the user to suit the device that will be used for output. Either 60- or 120-column plots can be produced for output on an 80- or 132-column printer. The plot length is fixed at 60 rows.

If a high-resolution plot is desired, high-resolution plot information is routed to a disk file which contains a "compressed plot data set." The file name is chosen by the user and has the form "*xxxxx.POP" where x represents alphanumeric characters and * represents an alphabetic character. This file is a concise representation of the plot information and consists of 50-150 card images per plot.

The compressed plot data may be viewed on a graphics terminal using PDP-10 system utility programs POPTEK or QWKPOP. Alternatively, the plot data may be routed to a remote Calcomp pen-and-ink plotter to obtain a high-resolution paper copy using the command "GRAPH C92:=xxxxxx.POP."

A third option consists of routing the plot data to a FR-80 film plotter for 35mm plots suitable for projection slides. This submission is performed by the command "GRAPH FR8:xxxxxx.POP." Before either of the GRAPH commands are given, the user must have created in his disk area a file having the name "SWITCH.INI". This file contains the user's name, address, and charge number. An example of a valid SWITCH.INI file is the following:

```
GRAPH/UID:DEF/NAME:"D. E. FIELDS"/PHONE:45435/ADRES:"BLDG7509"/CHARGE:
12345/FORMS:600
```

In the form presented in this report, the TERPED/P program is interactive and requires access to various DISSPLA subprograms, including the locally available postprocessor options under which the local and remote plotting commands discussed above are supported. With a few modifications, TERPED/P can be made independent of the DISSPLA requirements, but this also removes the high-resolution plot capability. The modifications are given in Appendix G.

4. DISCUSSION OF CODE USAGE

TERPED/P is interactive and produces many "prompting" messages that the user should find self explanatory. The code also generates a number of warning and error messages, some of which are abbreviated. These messages are summarized in Table 2.

Appendix A contains the computer-user dialog for a complete run of the TERPED/P program. In the sample dialog shown in this Appendix, the user first assigns his terminal the device number 6 by executing the command "ASS TTY 6." He then begins execution of the program and provides access to various plotting utilities with the command "EX TERPED.P, @ SYS:DISPOP". The SYS:DISPOP portion of the command permits linkage to DISSPLA subroutines mentioned in Chaps. 3 and 5. The user then uses the code to analyze a set of data. First, a normal hypothesis is selected with a chi-squared goodness-of-fit test. Next, a printer plot is generated to visually check the data and then a high-resolution compressed plot data set is generated. The user then restarts the code but selects the lognormal hypothesis. A printer plot is then generated for a visual check and finally a compressed plot data set is generated. The user then exits the code and checks the file directory for the plot data file (extension .POP). A text summary of the plot is generated concurrently with the plot data set and is automatically sent to a line printer and the disk file deleted upon exit of the program. The line printer listing of the plot summary is in Appendix B.

The last command in Appendix A routes a copy of the compressed plot data file (ENZYME.POP) to a remote Calcomp plotter. For this command to be executed, the user must have prepared a SWITCH.INI file as discussed in Chap. 3.

Appendix C documents the use of the POPTEK utility program to view a plot data file (ENZYME.POP) on a local Tektronix 4006 graphics terminal. Hard copy of the information displayed on the terminal screen was produced by a Tektronix 4361 hard copy plotter interfaced to the terminal.

Table 2. List of error messages generated by TERPED/P

Message	Comments
A) Number of points cannot exceed 300	TERPED arrays are dimensioned to accept up to 300 data points. The user must redimension for data sets of greater than 300 points.
B) Invalid input; try again	The user gave an inappropriate response to the prompt message.
C) Error from ICSICU, from DSPROB: Data Point J. ERR=I	<p>Evaluation of whether the cubic spline coefficients about the <u>J</u>th data point has been unsuccessful. For NX=number of points, less 1, and for IC=row dimension of spline coefficient matrix;</p> <p>ERR=129 indicates $IC < (NX-1)$. ERR=130 indicates $NX < 2$. ERR=131 indicates data points incorrectly ordered.</p> <p>For an evaluation of the severity of these errors, see E).</p>
D) Error from ISCEVU, from DSPROP; Data Point J, ERR=I	<p>Evaluation of the linearized probability for the <u>J</u>th data point has been unsuccessful.</p> <p>ERR=33 indicates the calculated abscissa of the cubic spline is less than the minimum abscissa represented in the data. ERR=34 indicates the calculated abscissa of the cubic spline is greater than the maximum abscissa represented in the data.</p> <p>For comments on the severity of these errors, see E).</p>

Table 2. (continued)

Message	Comments
E) ***IMSL(UERTST)*** (Warning type) Name IER=K	(Warning type) is either "warning," "warning with fix," "terminal" error, or "non-defined." NAME identifies the subroutine in which the error was generated. This message supplements C) and D). The value of K is computed, given I defined as in C) and D), as follows: K=I+32 (warning) =I+64 (warning with fix) =I+128 (terminal error).
F) ***IMSL(UERTST)*** (Warning type) GFIT IER=L	Calculation of Chi-squared statistic has been unsuccessful (Terminal error) L = 129 indicates error occurred in subroutine MDCDFI 130 indicates number of cells <2 (Warning) = 35 number of data points within cell <5 36 number of data points within cell <1, in which case significance level set to negative machine infinity
G) ***IMSL(UERTST)*** Warning NKS1 IER=33	"Confidence Level" of Kolmogorov-Smirnov one-sample test cannot be used because sample size is less than 80

Appendix D contains the high-quality plots produced by a Calcomp plotter as requested in Appendix A and previewed in Appendix C.

A FORTRAN listing of the TERPED/P code is included in Appendix E. Subroutines and functions are listed alphabetically, with the exception of the main program and block data, which are respectively listed first and second. As mentioned previously, there are many calls to DISSPLA subprograms; and these proprietary software are not included in the listing.

The data set used for the sample run of Appendix A is reproduced in Appendix F.

5. TERPED/P PROGRAM STRUCTURE

The TERPED/P program consists of 24 subprograms and a number of subroutines that comprise part of the DISSPLA graphics package. The subprogram structure is shown in Figure 1. Provision has been made for two methods of data entry. Statistics are output to the computer terminal as they are computed. Graphical outputs are user selectable and consist of a printer plotter and a high-resolution plotter.

A FORTRAN listing of TERPED/P is included in Appendix E. The Main Program structures flow of control. It also accepts some control information and performs intermediate calculations for computing the correlation between input data and the corresponding hypothetical distribution.

Subroutine DSINP prompts the user to input program control information and data. Data can be entered directly into the program by using the terminal keyboard or can be read from a file that has been stored on disk. The file must be named "FORxx.DAT" where xx is between 07 and 63, inclusively. The sample run in Appendix A documents use of a data file.

The data are ranked according to magnitude by subroutine DSORD. Subroutine DSPROB computes a cumulative probability for each data value. It also performs a cubic spline fit to a table of error function arguments stored in the BLOCK DATA subprogram to compute the corresponding

Figure 1. TERPED/P subprogram structure.

linearized cumulative probability. Subroutines ICSICU* and ICSEVU* are called by DSPROB to determine the spline coefficients near the data point and to evaluate the linearized probability, respectively.

Subroutine DSANAL performs preliminary data analysis and if the user has called for a chi-squared test or KS test, calls GFIT* or NKS1*, respectively. GFIT uses subroutine MDCDFI* to compute the probability for rejection of the null hypothesis and in turn calls ERF* to compute the value of the error function. NKS1 invokes MDSMR which computes the "confidence level" for the KS test assuming a large number of points ($m > 80$). Both GFIT and NKS1 use subroutine DSNORM which is the theoretical probability distribution function against which the data are tested. All of the IMSL subroutines call subroutine UERTST which generates and prints warning and error messages which are discussed in Chapter 4.

Subroutine DSFIT performs a linear least-squares fit to the data points. Subroutine ALSQ (Westley and Watts, 1970), which determines the coefficients of the fit, is called by DSFIT.

Pierson's correlation coefficient between data points and corresponding points along the linear least-squares fit gives a measure of the validity of normal or lognormal hypothesis. The correlation coefficient is calculated by subroutine DSCOR, but is printed by the main program.

Subroutine PRNCON initializes a matrix containing printer-plotter information and calls subroutine LINPLT to plot data according to a normal hypothesis or subroutine LOGPLT, which plots data when the log-normal hypothesis has been selected. PRNCON, LINPLT, and LOGPLT use subroutine NORM to normalize data according to user input scaling values. Subroutine PRNPLT outputs the plot to the user-specified device. The 'F' character represents the fitted point, the 'D' character represents a data point, and a '*' represents fit and data point coincidence.

Subroutine DSPLOT plots the input data and the results of the linear least-squares fit on a user-defined ordinate with a cumulative probability-derived abscissa. The plot is high quality/resolution and

*Software written by International Math and Statistical Libraries, Houston, Texas.

can be routed to a Calcomp paper plotter or FR80 film plotter. DSPL0T
uses several DISSPLA subroutines

REFERENCES

- Aitchison, J., and Brown, J. 1969. *The Lognormal Distribution*, Cambridge Press, New York.
- Ahlberg, J., Nelson, E., and Walsh, J. 1967. *The Theory of Splines and Their Applications*. Academic Press, New York.
- Bradley, J. V. 1968. *Distribution-Free Statistical Tests*. Prentice-Hall, Inc., New Jersey. pp. 367-69.
- Fields, D. E., Little, C. A., and Shaeffer, D. L. 1980. *User's Manual for TEDPED, An Interactive Code for Testing the Hypothesis of Log-normally Distributed Data*, Oak Ridge National Laboratory Report, ORNL/TM-6933.
- Fields, D. E. 1981. *TERPED: A Versatile Code for Examining the Distribution of Experimental Data*, Oak Ridge National Laboratory Report, ORNL-5689.
- Gosslee, D. G. Statistics Department, Mathematics Division, Oak Ridge National Laboratory, Personal Communication, June 12, 1972.
- Kendal, M. G., and Stuart, A. 1961. *The Advanced Theory of Statistics*, Vol. 2, Hafner Publishing Co., New York.
- Kimball, B. F. "On the Choice of Plotting Positions on Probability Paper," *American Statistical Association Journal*, September, 1960.
- Siegel, S. 1956. *Nonparametric Statistics for the Behavioral Sciences*, McGraw-Hill Publishing Company, New York.
- Snedecor, G. W., and Cochran, W. G. 1972. *Statistical Methods* (Sixth Ed.). Iowa State University Press, Ames, Iowa.
- Westley, G. W., and Watts, J. S. (Eds.) 1970. *The Computing Technology Center Numerical Analysis Library*. Report No. CTC-39, Union Carbide Corporation, Nuclear Division, Oak Ridge, Tennessee.

APPENDIX A

Sample Run

**THIS PAGE
WAS INTENTIONALLY
LEFT BLANK**

APPENDIX A. Sample Run

This appendix consists of a sample TERPED/P run showing analyses of experimental data consisting of 22 points (shown in Appendix F) and the generation of two printer plots and two high-resolution plots stored in a compressed data file called ENZYME.POP. The first analysis and plot assumes a normal distribution of experimental data. A chi-squared test is included in the first analysis. In the second analysis, the same data are used but a lognormal distribution is assumed. Output files produced are ENZYME.POP (the compressed plot data file) and SUM.PLT (plot summary file). Also, the command which submits the compressed data file to the Calcomp plotter is shown.

ASS TTY 6
TTY31 assigned

.EX TERPED.P,@SYS:DISPOP
LINK: Loading
[LNKXCT TERPED execution]

CODE TERPED/P, VERSION 1/81

LINEAR OR LOG-NORMAL ANALYSIS? (3 CHAR)
LIN
PERFORM KSL TEST? (YES OR NO)
NO
AND HOW ABOUT A CHI SQUARED TEST? (YES OR NO)
YES
INPUT TITLE OF PLOT, UP TO 19 CHAR.
ENZYME LEVEL
EXP. DATA INPUT: HAND INPUT OR FILE? (4 CHAR)
FILE
ENTER INPUT FILE NUMBER
4
ENTER NUMBER OF DATA POINTS TO BE USED.
(IF NUMBER = 0, ALL POINTS USED FOR FILE INPUT)
0
DATA INPUT COMPLETE; ANALYSIS STARTED.
NUMBER OF POINTS = 22
PRINT ORDERED DATA (YES OR NO)?
YES

J	VALUE	PROB
1	0.288E+02	0.0281
2	0.293E+02	0.0730
3	0.297E+02	0.1180
4	0.370E+02	0.1629
5	0.373E+02	0.2079
6	0.421E+02	0.2528
7	0.454E+02	0.2978
8	0.460E+02	0.3427
9	0.460E+02	0.3876
10	0.477E+02	0.4326
11	0.479E+02	0.4775
12	0.496E+02	0.5225
13	0.511E+02	0.5674

14	0.566E+02	0.6124
15	0.573E+02	0.6573
16	0.586E+02	0.7022
17	0.588E+02	0.7472
18	0.606E+02	0.7921
19	0.609E+02	0.8371
20	0.622E+02	0.8820
21	0.622E+02	0.9270
22	0.704E+02	0.9719

DATA HAS BEEN RANK ORDERED

LIN NORMAL ANALYSIS:

MEAN = 4.934E+01 AND S.D. = 1.181E+01

INPUT NO. OF CELLS FOR CHI SQUARED TEST (INTEGER)

4

BEGIN GFIT

CHI SQUARED STATISTIC = 0.9091

NULL HYPOTHESIS WOULD BE REJECTED AT 0.340 LEVEL

BEGIN LEAST SQUARES DATA FIT.

COEFFICIENTS OF FIT ARE 0.120E+02 0.493E+02

CORRELATION BETWEEN DATA AND FIT IS R= 0.979

ENTER H TO GENERATE PLOT DATA SET, L FOR PRINTER PLOT,
R FOR RESTART, E FOR EXIT.

L

ENTER Y-AXIS LABEL (20 CHARACTERS MAXIMUM)

I.U./LITER

ENTER NUMBER OF COLUMNS FOR PLOT (60 OR 120)

120

ENTER OUTPUT DEVICE (TTY, DSK, OR LPT)

TTY

ENTER Y-AXIS ORIGIN

BETWEEN -1E8 AND 1E8

20

ENTER Y-AXIS STEP SIZE (UNITS/5 LINES OF PLOT,
60 LINES OF PLOT)

5

ENTER H TO GENERATE PLOT DATA SET, L FOR PRINTER PLOT,
R FOR RESTART, E FOR EXIT.
H

ENTER VERTICAL AXIS LABEL
I.U./LITER
DEFINED ORDINATE LENGTH IS 5 INCHES...
ENTER Y ORIGIN.

25
ENTER Y AXIS STEP SIZE (UNITS/INCH).

10
ENTER NAME FOR POP FILE.
(EXTENSION .POP WILL BE ADDED)

ENZYME
ENTER H TO GENERATE PLOT DATA SET, L FOR PRINTER PLOT,
R FOR RESTART, E FOR EXIT.
R

CODE TERPED/P, VERSION 1/81

LINEAR OR LOG-NORMAL ANALYSIS? (3 CHAR)
LOG

PERFORM KSL TEST? (YES OR NO)
NO

AND HOW ABOUT A CHI SQUARED TEST? (YES OR NO)
NO

INPUT TITLE OF PLOT, UP TO 19 CHAR.
ENZYME LEVEL

EXP. DATA INPUT: HAND INPUT OR FILE? (4 CHAR)
FILE

ENTER INPUT FILE NUMBER

4
ENTER NUMBER OF DATA POINTS TO BE USED.
(IF NUMBER = 0, ALL POINTS USED FOR FILE INPUT)

0
DATA INPUT COMPLETE; ANALYSIS STARTED.
NUMBER OF POINTS = 22
PRINT ORDERED DATA (YES OR NO)?

YES

J	VALUE	PROB
1	0.336E+01	0.0281
2	0.338E+01	0.0730
3	0.339E+01	0.1180
4	0.361E+01	0.1629
5	0.362E+01	0.2079
6	0.374E+01	0.2528
7	0.382E+01	0.2978
8	0.383E+01	0.3427
9	0.383E+01	0.3876
10	0.386E+01	0.4326
11	0.387E+01	0.4775
12	0.390E+01	0.5225
13	0.393E+01	0.5674
14	0.404E+01	0.6124
15	0.405E+01	0.6573
16	0.407E+01	0.7022
17	0.407E+01	0.7472
18	0.410E+01	0.7921
19	0.411E+01	0.8371
20	0.413E+01	0.8820
21	0.413E+01	0.9270
22	0.425E+01	0.9719

DATA HAS BEEN RANK ORDERED

LOG NORMAL ANALYSIS:

MEAN = 3.868E+00 AND S.D. = 2.606E-01

MOST PROBABLE VALUE = 4.471E+01

MEDIAN VALUE = 4.786E+01

MEAN VALUE = 4.951E+01

99% QUANTILE = 8.775E+01

BEGIN LEAST SQUARES DATA FIT.

COEFFICIENTS OF FIT ARE 0.259E+00 0.387E+01

CORRELATION BETWEEN DATA AND FIT IS R= 0.957

ENTER H TO GENERATE PLOT DATA SET, L FOR PRINTER PLOT,
R FOR RESTART, E FOR EXIT.

L

ENTER Y-AXIS LABEL (20 CHARACTERS MAXIMUM)
I.U./LITER

ENTER NUMBER OF COLUMNS FOR PLOT (60 OR 120)
120

ENTER OUTPUT DEVICE (TTY, DSK, OR LPT)
TTY

ENTER Y-AXIS ORIGIN

BETWEEN 1E-15 AND 1E8
(CANNOT BE ZERO)
10

ENTER NUMBER OF CYCLES FOR LOG PLOT (INTEGER, 1-5)
1

ENTER H TO GENERATE PLOT DATA SET, L FOR PRINTER PLOT,
R FOR RESTART, E FOR EXIT.

H

ENTER VERTICAL AXIS LABEL

I.U./LITER

DEFINED ORDINATE LENGTH IS 5 INCHES...

ENTER Y ORIGIN (NOT ZERO).

10

ENTER NUMBER OF INCHES PER LOG CYCLE.

1\1\5

ENTER H TO GENERATE PLOT DATA SET, L FOR PRINTER PLOT,
R FOR RESTART, E FOR EXIT.

E

END OF DISSPLA 8.2 -- 4642 VECTORS GENERATED IN 2 PLOT FRAMES.
-ISSCO- 4186 SORRENTO VALLEY BLVD., SAN DIEGO CALIF. 92121

DISSPLA IS A CONFIDENTIAL PROPRIETARY PRODUCT OF ISSCO AND ITS USE
IS SUBJECT TO A NONDISSEMINATION AND NONDISCLOSURE AGREEMENT.

STOP

END OF EXECUTION

CPU TIME: 3.17 ELAPSED TIME: 2:19.60

EXIT

*DIR *.POP

ENZYME POP 32 <055> 20-APR-81 DSKB: [6642,17]

.GRA C92:=ENZYME.POP

JOB QUEUED FOR THE 360/

APPENDIX B

Sample DISSPLA Plot Summary

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

APPENDIX B. Sample DISSPLA Plot Summary

This file is stored as SUM.PLT in the user's disk area while the program is running as shown in Appendix A. When the user exits the program, the file is immediately queued for the line printer and deleted from the disk. The file is not created if no high-resolution plots are made.

PLOT NO. 1 WITH THE TITLE
 ENZYME LEVEL
 HAS BEEN COMPLETED.

PLOT ID. READS
 PLOT 1 15.53.00 MON 20 APR, 1981
 , ORNL DISSPLA VER 8.2

DATA FOR PLOT

NO. OF CURVES DRAWN 2

 HORIZ. AXIS LENGTH 8.0 INS.
 VERT. AXIS LENGTH 5.0 INS.

HORIZ. ORIGIN 0.1000E+01
 VERT. ORIGIN 0.2500E+02

HORIZ. AXIS LINEAR
 STEP SIZE 0.1500E+01 UNITS/INCH

VERT. AXIS LINEAR
 STEP SIZE 0.1000E+02 UNITS/INCH

.....
 . LOCATION OF CURRENT PHYSICAL ORIGIN .
 . X= 1.00 Y= 1.14 INCHES .
 . FROM LOWER LEFT CORNER OF PAGE .

PLOT NO. 2 WITH THE TITLE
 ENZYME LEVEL
 HAS BEEN COMPLETED.

PLOT ID. READS

PLOT 2 16.00.00 MON 20 APR, 1981
 , CRNL DISSPLA VER 8.2

DATA FOR PLOT

NO. OF CURVES DRAWN 2

 HORIZ. AXIS LENGTH 8.0 INS.

VERT. AXIS LENGTH 5.0 INS.

HORIZ. ORIGIN 0.1000E+01

VERT. ORIGIN 0.1000E+02

HORIZ. AXIS LINEAR

STEP SIZE 0.1500E+01 UNITS/INCH

VERT. AXIS LOG

CYCLE LENGTH 5.00 INS./CYCLE

.....
 . LOCATION OF CURRENT PHYSICAL ORIGIN .
 . X= 1.00 Y= 1.14 INCHES .
 . FROM LOWER LEFT CORNER OF PAGE .

**THIS PAGE
WAS INTENTIONALLY
LEFT BLANK**

APPENDIX C

Preview of Compressed Plot Data File

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

R POPTEK

DISSPLA INTERMEDIATE PLOT FILE POSTPROCESSOR
ENTER SPECIAL REQUESTS OR PRESS -RETURN-

ENTER NAME FOR POP FILE.
(EXTENSION .POP WILL BE ADDED)
ENZYME

**THIS PAGE
WAS INTENTIONALLY
LEFT BLANK**

APPENDIX D

High-Resolution Plot Produced by TERPED/P

**THIS PAGE
WAS INTENTIONALLY
LEFT BLANK**

APPENDIX D. High-Resolution Plot Produced by TERPED/P

The plots were generated using the compressed data file previewed in Appendix C. The command requesting the plots was GRA C92:=ENZYME.POP as shown in Appendix A. Successful plot submission requires presence of a user identification file (SWITCH.INI) or its equivalent, as discussed in Chapter 3.

ORNL-DWG 81-10030

**THIS PAGE
WAS INTENTIONALLY
LEFT BLANK**

APPENDIX E

FORTRAN-10 Listing of TERPED/P Code

**THIS PAGE
WAS INTENTIONALLY
LEFT BLANK**

APPENDIX E. FORTRAN-10 Listing of TERPED/P Code

This version of the TERPED/P code executes on the Digital Equipment Corporation PDP-10 computer system. Utilities required are the DISSPLA graphics package with the post-processor option for generation of compressed plot data sets. FORTRAN-10 is the extended version of FORTRAN-IV that is available on the PDP-10.

TERPED/P is a revised version of the TERPED code documented in Oak Ridge National Laboratory Report ORNL-5689.

```

CTERPED/P
CODE, TERPED BY D.E. FIELDS, C.A. LITTLE AND D.L. SHAEFFER
CODE, TERPED, A MODIFICATION BY D.E. FIELDS
CODE, TERPED/P, A MODIFICATION BY S.R. GLANDON
CODE MODIFIED 12/80-1/81
C      ANALYSES AND PLOTS NORMALLY OR LOG-NORMALLY DISTRIBUTED DATA
C      AND RESULTS OF LEAST SQUARES DATA FIT ON DATA
C      ALSO PROVIDES STATISTICAL PARAMETERS DESCRIBING EXPERIMENTAL DATA
DIMENSION DATA(300), IORD(300), PARM(2,300), B(2), TITLE(5), ICOR(2),
1 Y(300), Y1(300)
DATA DDL/' ' //, E/'E'//, P/'P'//, PL/'H'//, IPL/0//, AL/'L'//
CALL COMPRS
10 DO 20 I=1,5
20   TITLE(I)=DDL
   WRITE(6,10000)
C   SPECIFY INPUT DATA
CALL DSINP(DATA, IYPE, NP, TITLE, IV, ICOP)
IF (IYPE.EQ.1) GO TO 30
ON 30 I=1, NP
   DATA(I)=ALOG(DATA(I))
30   CONTINUE
C   ORDER DATA
CALL DSORD(DATA, NP, IORD)
C   COMPUTE PROBABILITIES AND A-VALUE
CALL DSPROB(DATA, IYPE, NP, IORD, PARM)
WRITE(6,10100)
C   COMPUTE AND OUTPUT MEAN AND STANDARD DEVIATION
CALL DSANAL(DATA, IYPE, NP, ICOR)
C   PERFORM LEAST SQUARES FIT ON DATA
CALL DSFIT(PARM, NP, B)
C   COMPUTE CORRELATION COEFF.
DO 40 N=1, NP
   Y(N)=PARM(1, N)
   Y1(N)=B(1)*PARM(2, N)+B(2)
   IF (IYPE.EQ.1) GO TO 40
   Y(N)=EXP(Y(N))
   Y1(N)=EXP(Y1(N))
40   CONTINUE
CALL DSCOR(Y, Y1, NP, COR)
WRITE(6,10200) COR
50 WRITE(6,10300)
READ(6,10400) A
IF (A.EQ.E) GO TO 70
IF (A.EQ.AL) GO TO 60
IF (A.EQ.P) GO TO 10
IF (A.NE.P) GO TO 50
C   PLOT DATA AND RESULTS OF FIT
IF (IPL.EQ.0) OPEN(UNIT=1, DEVICE='DSK', FILE='SUM.PLT')
IPL=IPL+1
CALL DSPLT(B, NP, PARM, IYPE, IV, TITLE, COR, IPL)
GO TO 50
60 CALL PRNCON(PARM, B, NP, IYPE, TITLE)
GO TO 50
70 IF (IPL.EQ.0) GO TO 80
CALL SETDEV(6, 6)
CALL DONEPL
CLOSE(UNIT=1, DISPOSE='LIST')
80 STOP
10000 FORMAT(2X, /, 2X, 'CODE TERPED/P, VERSION 1/81', //)
10100 FORMAT(2X, 'DATA HAS BEEN RANK ORDERED')
10200 FORMAT(2X, 'CORRELATION BETWEEN DATA AND FIT IS R=', F6.3, //)
10300 FORMAT(2X, 'ENTER H TO GENERATE PLOT DATA SET, L FOR PRINT',
1 'ER PLOT, /, /, ' R FOR RESTART, E FOR EXIT.')

```

10400 FORMAT(A11
END

TERP 315
TERP 320

CBLOCK

BLOCK DATA

C D.E. FIELDS, C.A. LITTLE, AND D.L. SHAEFFER

CODE, TERP, A MODIFICATION BY D.E. FIELDS

C DATA USED BY DATA ANALYSIS PACKAGE

COMMON AP(2,101),DSUM,RMS

DATA AP/-5.00000000, 0.00000028,-4.90000000, 0.00000048,

E-4.80000000, 0.00000079,-4.70000000, 0.00000130,
E-4.60000000, 0.00000212,-4.50000000, 0.00000340,
E-4.40000000, 0.00000541,-4.30000000, 0.00000854,
E-4.20000000, 0.00001334,-4.10000000, 0.00002066,
E-4.00000010, 0.00003167,-3.90000000, 0.00004809,
E-3.80000000, 0.00007235,-3.70000000, 0.00010780,
E-3.60000000, 0.00015911,-3.50000000, 0.00023263,
E-3.40000000, 0.00033693,-3.30000000, 0.00048342,
E-3.20000000, 0.00068714,-3.10000000, 0.00096761,
E-3.00000000, 0.00134990,-2.90000000, 0.00186581,
E-2.80000000, 0.00255513,-2.70000000, 0.00346697,
E-2.60000000, 0.00466119,-2.50000000, 0.00620966,
E-2.40000000, 0.00819754,-2.30000000, 0.01072411,
E-2.20000000, 0.01390345,-2.10000000, 0.01786442,
E-2.00000000, 0.02275013,-1.90000000, 0.02871656,
E-1.80000000, 0.03593032,-1.70000000, 0.04456546,
E-1.60000000, 0.05479930,-1.50000000, 0.06680720,
E-1.40000000, 0.08075666,-1.30000000, 0.09680048,
E-1.20000000, 0.11506967,-1.10000000, 0.13566606,
E-1.00000010, 0.15865525,-0.90000004, 0.18406012,
E-0.80000001, 0.21185540,-0.70000005, 0.24196364,
E-0.60000002, 0.27425312,-0.50000006, 0.30853752,
E-0.40000004, 0.34457825,-0.30000001, 0.38208858,
E-0.20000005, 0.42074028,-0.10000002, 0.46017216,
E-0.00000006, 0.49999999, 0.09999996, 0.53982782,
E 0.19999993, 0.57925967, 0.29999995, 0.61791141,
E 0.39999998, 0.65542173, 0.49999994, 0.69146244,
E 0.59999996, 0.72574686, 0.69999993, 0.75803622,
E 0.79999995, 0.78814459, 0.89999998, 0.81593987,
E 0.99999994, 0.84134473, 1.10000000, 0.86433393,
E 1.19999990, 0.88493031, 1.30000000, 0.90319951,
E 1.40000000, 0.91924334, 1.49999990, 0.93319279,
E 1.60000000, 0.94520070, 1.69999990, 0.95543453,
E 1.80000000, 0.96406968, 1.90000000, 0.97128344,
E 1.99999990, 0.97724986, 2.10000000, 0.98212558,
E 2.19999990, 0.98609655, 2.30000000, 0.98927589,
E 2.40000000, 0.99180246, 2.49999990, 0.99379034,
E 2.60000000, 0.99533881, 2.69999990, 0.99653303,
E 2.80000000, 0.99744487, 2.90000000, 0.99813419,
E 3.00000000, 0.99865010, 3.09999990, 0.99903240,
E 3.19999990, 0.99931286, 3.30000000, 0.99951658,
E 3.40000000, 0.99966307, 3.50000000, 0.99976737,
E 3.59999990, 0.99984089, 3.69999990, 0.99989221,
E 3.80000000, 0.99992765, 3.90000000, 0.99995191,

```

E 4.00000000, 0.99996833, 4.09999990, 0.99997935,
E 4.19999990, 0.99998666, 4.30000000, 0.99999146,
E 4.40000000, 0.99999459, 4.50000000, 0.99999660,
E 4.59999990, 0.99999789, 4.69999990, 0.99999870,
E 4.80000000, 0.99999921, 4.90000000, 0.99999952,
E 4.99999990, 0.99999972/
END

```

```

SUBROUTINE ALSQ(A,Y,B,P2,NN,MM,NA)
  IMPLICIT DOUBLE PRECISION (A-H,O-Z)
  DIMENSION A(NA,3),Y(300),B(2)
  N = NN
  N1 = N+1
  M = MM
  M1 = M+1
  MM1 = M-1

  REDUCE THE LEAST SQUARES MATRIX TO UPPER TRIANGULAR FORM

  DO 60 L=1,M
    SS = 0.
    DO 10 I=L,N
      SS = SS + A(I,L)**2
    S2 = SS
    S = SQRT(S2)
    IF (A(L,L).LT.0.) S=-S
    D = S2 + S*A(L,L)
    A(L,L) = A(L,L) + S
    IF (L.EQ.M) GO TO 50
    L1 = L+1
    DO 30 J=L1,M
      PP = 0.
      DO 20 I=L,N
        PP = PP + A(I,L)*A(I,J)
      A(N1,J) = PP/D
    DO 40 J=L1,M
      DO 40 I=L,N
        A(I,J) = A(I,J) - A(I,L)*A(N1,J)
    A(N1,L) = -S
    60 CONTINUE
  GO TO 70

  REDUCE THE VECTOR Y

  M = MM
  MM1 = M-1
  70 DO 80 I=1,N
    80 A(I,M1) = Y(I)
    DO 100 L=1,M
      PP = 0.
      DO 90 I=L,N
        PP = PP + A(I,L)*A(I,M1)
      D = PP/(-A(L,L)*A(N1,L))
      DO 100 I=L,N

```

ALSQ	0
ALSQ	5
ALSQ	10
ALSQ	15
ALSQ	20
ALSQ	25
ALSQ	30
ALSQ	35
ALSQ	40
ALSQ	45
ALSQ	50
ALSQ	55
ALSQ	60
ALSQ	65
ALSQ	70
ALSQ	75
ALSQ	80
ALSQ	85
ALSQ	90
ALSQ	95
ALSQ	100
ALSQ	105
ALSQ	110
ALSQ	115
ALSQ	120
ALSQ	125
ALSQ	130
ALSQ	135
ALSQ	140
ALSQ	145
ALSQ	150
ALSQ	155
ALSQ	160
ALSQ	165
ALSQ	170
ALSQ	175
ALSQ	180
ALSQ	185
ALSQ	190
ALSQ	195
ALSQ	200
ALSQ	205
ALSQ	210
ALSQ	215
ALSQ	220
ALSQ	225

```

100      A(I,M1) = A(I,M1) - D*A(I,L)
C
C      CALCULATE THE COEFFICIENT VECTOR B
      B(M) = A(M,M1)/A(N1,M)
      IF (M.EQ.1) GO TO 130
      DO 120 LL=1,MM1
        L = M-LL
        LL = L+1
        PP = A(L,M1)
        DO 110 I=L1,M
          PP = PP - A(L,I)*B(I)
110      PP = PP - A(L,I)*B(I)
120      B(L) = PP/A(N1,L)
C
C      CALCULATE R2
130      SS = 0.
      MPI = M+1
      DO 140 I=MPI,N
        SS = SS + A(I,M1)**2
140      A(I,M1) = 0.
      R2 = SS
C
C      PERFORM THE BACK CALCULATIONS
      DO 170 LL=1,M
        L = M-LL+1
        PP = 0.
        DO 150 I=L,N
          PP = PP + A(I,L)*A(I,M1)
150      D = PP/(-A(L,L)*A(N1,L))
        DO 160 I=L,N
          A(I,M1) = A(I,M1) - D*A(I,L)
160      A(I,M1) = A(I,M1) - D*A(I,L)
170      CONTINUE
      RETURN
      END

```

ALSQ 230
 ALSQ 235
 ALSQ 240
 ALSQ 245
 ALSQ 250
 ALSQ 255
 ALSQ 260
 ALSQ 265
 ALSQ 270
 ALSQ 275
 ALSQ 280
 ALSQ 285
 ALSQ 290
 ALSQ 295
 ALSQ 300
 ALSQ 305
 ALSQ 310
 ALSQ 315
 ALSQ 320
 ALSQ 325
 ALSQ 330
 ALSQ 335
 ALSQ 340
 ALSQ 345
 ALSQ 350
 ALSQ 355
 ALSQ 360
 ALSQ 365
 ALSQ 370
 ALSQ 375
 ALSQ 380
 ALSQ 385
 ALSQ 390
 ALSQ 395
 ALSQ 400
 ALSQ 405

```

SUBROUTINE DSANAL(DATA, ITYPE, NP, ICOR)
C
C      D.E. FIELDS, C.A. LITTLE, AND D.L. SHAEFFER
CODE. TERPED, A MODIFICATION BY D.E. FIELDS
      COMMON AP(2,101), DSUM, RMS
      EXTERNAL DSNORM
      DIMENSION DATA(300), ICOR(2), CELLS(10), COMP(10), PDIF(6)
      REAL MEDIAN, MOST, MEAN
      DSUM=0.0
      RMS=0.0
      IF (ITYPE.EQ.2) WRITE(6,10000)
      IF (ITYPE.EQ.1) WRITE(6,10100)
C
C      MEAN AND STANDARD DEVIATION
      DO 10 I=1,NP
10      DSUM=DSUM+DATA(I)
      DSUM=DSUM/NP
      DO 20 I=1,NP
20      RMS=RMS+(DSUM-DATA(I))**2

```

DSAN 0
 DSAN 5
 DSAN 10
 DSAN 15
 DSAN 20
 DSAN 25
 DSAN 30
 DSAN 35
 DSAN 40
 DSAN 45
 DSAN 50
 DSAN 55
 DSAN 60
 DSAN 65
 DSAN 70
 DSAN 75
 DSAN 80

```

RMS=SQRT(RMS/(NP-1))
SIGSQ=RMS*RMS
WRITE(6,10200)DSUM,RMS
IF (ITYPE.NE.2) GO TO 30
MEDIAN=EXP(DSUM)
MEAN=MEDIAN*EXP(SIGSQ/2.)
MOST=MEDIAN*EXP(-SIGSQ)
X99=MEDIAN*EXP(2.326*RMS)
WRITE(6,10300)MOST
WRITE(6,10400)MEDIAN
WRITE(6,10500)MEAN
WRITE(6,10600)X99
30 CONTINUE
IF (ICOR(2).NE.1) GO TO 40
WRITE(6,10700)
READ(6,10800)K
IDF=2
CALL GFI(DSNORM,K,DATA,NP,CELLS,COMP,CS,IDF,0,IER)
WRITE(6,10900)CS
WRITE(6,11000)Q
40 CONTINUE
IF (ICOR(1).NE.1) GO TO 50
WRITE(6,11100)
CALL NKSI(DSNORM,DATA,NP,POIF,IER)
IF (IER.EQ.33) WRITE(6,11200)
WRITE(6,11300)POIF(4),POIF(6)
WRITE(6,11400)
50 CONTINUE
RETURN
10000 FORMAT(/,2X,'LOG NORMAL ANALYSIS:')
10100 FORMAT(/,2X,'LIN NORMAL ANALYSIS:')
10200 FORMAT(2X,'MEAN = ',1PE11.3,' AND S.D. = ',F9.3)
10300 FORMAT(' MOST PROBABLE VALUE = ',T30,1PE14.3)
10400 FORMAT(' MEDIAN VALUE = ',T30,1PE14.3)
10500 FORMAT(' MEAN VALUE = ',T30,1PE14.3)
10600 FORMAT(' 99% QUANTILE = ',T30,1PE14.3)
10700 FORMAT(/,2X,'INPUT NO. OF CELLS FOR CHI SQUARED TEST (INTEGER)')
10800 FORMAT(I2)
10900 FORMAT(' CHI SQUARED STATISTIC = ',F8.4)
11000 FORMAT(' NULL HYPOTHESIS WOULD BE REJECTED AT',F6.3,' LEVEL')
11100 FORMAT(' BEGIN KSL TEST')
11200 FORMAT(' ASYMPTOTIC APPROXIMATION FOR N>80:')
11300 FORMAT(7X,'Z = ',F8.4,' AT A CONFIDENCE LEVEL OF ',F8.3)
11400 FORMAT(/,2X)
END

```

```

DSAN 85
DSAN 90
DSAN 95
DSAN 100
DSAN 105
DSAN 110
DSAN 115
DSAN 120
DSAN 125
DSAN 130
DSAN 135
DSAN 140
DSAN 145
DSAN 150
DSAN 155
DSAN 160
DSAN 165
DSAN 170
DSAN 175
DSAN 180
DSAN 185
DSAN 190
DSAN 195
DSAN 200
DSAN 205
DSAN 210
DSAN 215
DSAN 220
DSAN 225
DSAN 230
DSAN 235
DSAN 240
DSAN 245
DSAN 250
DSAN 255
DSAN 260
DSAN 265
DSAN 270
DSAN 275
DSAN 280
DSAN 285
DSAN 290
DSAN 295
DSAN 300
DSAN 305

```

```

SUBROUTINE DSCOR(X,Y,N,COR)
C   CALLED FROM DSANAL OF DATA STATISTICS PKG.
C   BY D.E. FIELDS, C.A. LITTLE, AND D.L. SHAEFFER
CODE, TERPED, A MODIFICATION BY D.E. FIELDS
DIMENSION X(300),Y(300)
XSUM=0.
YSUM=0.
XYSUM=0.

```

```

DSCD 0
DSCD 5
DSCD 10
DSCD 15
DSCD 20
DSCD 25
DSCD 30
DSCD 35

```

```

X2SUM=0.
Y2SUM=0.
DO 10 I=1,N
  XSUM = XSUM + X(I)
  YSUM = YSUM + Y(I)
10  CONTINUE
YBAR = YSUM / N
XBAR = XSUM / N
DO 20 I=1,N
  XYSUM = XYSUM + ((X(I)-XBAR)*(Y(I)-YBAR))
  X2SUM = X2SUM + (X(I)-XBAR)**2
  Y2SUM = Y2SUM + (Y(I)-YBAR)**2
20  CONTINUE
COR = XYSUM / SORT(X2SUM*Y2SUM)
RETURN
END

```

DSCO 40
 DSCO 45
 DSCO 50
 DSCO 55
 DSCO 60
 DSCO 65
 DSCO 70
 DSCO 75
 DSCO 80
 DSCO 85
 DSCO 90
 DSCO 95
 DSCO 100
 DSCO 105
 DSCO 110
 DSCO 115

```

SUBROUTINE DSFIT(PARM,NP,B)
C   CALLED BY DATA STATISTICS PKG.
C   D.E. FIELDS, C.A. LITTLE, AND D.L. SHAEFFER
CODE, TERPED, A MODIFICATION BY D.E. FIELDS
DIMENSION PARM(2,300),A(301,3),Y(300),B(2)
C   PARM CONTAINS RANK ORDERED DATA ELEMENTS AND 'A' VALUE
C   FIT TO LINEAR FUNCTION PARM(1,N)=B(1)*PARM(2,N)+B(2)
DO 10 N=1,NP
  Y(N)=PARM(1,N)
  A(N,1)=PARM(2,N)
  A(N,2)=1.
10  CONTINUE
N=NP
WRITE(6,10000)
CALL ALSQ(A,Y,B,P2,N,2,301)
WRITE(6,10100)(B(NC),NC=1,2)
RETURN
10000 FORMAT(2X,'BEGIN LEAST SQUARES DATA FIT.',/)
10100 FORMAT(2X,'COEFFICIENTS OF FIT ARE',2E10.3)
END

```

DSFI 0
 DSFI 5
 DSFI 10
 DSFI 15
 DSFI 20
 DSFI 25
 DSFI 30
 DSFI 35
 DSFI 40
 DSFI 45
 DSFI 50
 DSFI 55
 DSFI 60
 DSFI 65
 DSFI 70
 DSFI 75
 DSFI 80
 DSFI 85
 DSFI 90
 DSFI 95

```

SUBROUTINE DSINP(DATA,ITYPE,NP,TITLE,IV,ICOR)
C   ACCEPTS OR SPECIFIES INPUT FOR DATA STATISTICS PGM
C   D.E. FIELDS, C.A. LITTLE, AND D.L. SHAEFFER
CODE, TERPED, A MODIFICATION BY D.E. FIELDS
DIMENSION DATA(300),TITLE(5),ICOP(2)
DATA H/'HAND'/,F/'FILE'/,LIN/'LIN'/,LOG/'LOG'/
DATA MES/'YES'/,NO/'NO'/

```

DSIN 0
 DSIN 5
 DSIN 10
 DSIN 15
 DSIN 20
 DSIN 25
 DSIN 30

ICOR(1)=0	DSIN 35
ICOR(2)=0	DSIN 40
C SET DATA TYPE, LINEAR OR LOG-NORMAL	DSIN 45
10 WRITE(6,10000)	DSIN 50
READ(6,10100)LT	DSIN 55
IF (LT.NE.LIN) GO TO 20	DSIN 60
!TYPE=1	DSIN 65
GO TO 40	DSIN 70
20 IF (LT.EQ.LOG) GO TO 30	DSIN 75
WRITE(6,10200)	DSIN 80
GO TO 10	DSIN 85
30 !TYPE=2	DSIN 90
40 CONTINUE	DSIN 95
C SPECIFY FITTING TEST	DSIN 100
WRITE(6,10300)	DSIN 105
50 READ(6,10100)IL	DSIN 110
IF (IL.NE.MES) GO TO 60	DSIN 115
ICOR(1)=1	DSIN 120
GO TO 70	DSIN 125
60 IF (IL.EQ.NO) GO TO 70	DSIN 130
WRITE(6,10200)	DSIN 135
GO TO 50	DSIN 140
70 WRITE(6,10400)	DSIN 145
80 READ(6,10100)IL	DSIN 150
IF (IL.NE.MES.AND.L.NE.NO) WRITE(6,10200)	DSIN 155
IF (IL.NE.MES.AND.L.NE.NO) GO TO 80	DSIN 160
IF (IL.EQ.MES) ICOR(2)=1	DSIN 165
C SPECIFY ORDINATE TYPE	DSIN 170
90 CONTINUE	DSIN 175
IF (LT.NE.LIN) GO TO 100	DSIN 180
IV=1	DSIN 185
GO TO 120	DSIN 190
100 IF (LT.EQ.LOG) GO TO 110	DSIN 195
WRITE(6,10200)	DSIN 200
GO TO 90	DSIN 205
110 IV=2	DSIN 210
120 CONTINUE	DSIN 215
C SPECIFY TITLE	DSIN 220
WRITE(6,10500)	DSIN 225
READ(6,10600)ITITLE(I),I=1,4	DSIN 230
C SPECIFY INPUT DATA ORIGIN	DSIN 235
130 WRITE(6,10700)	DSIN 240
READ(6,10800)IA	DSIN 245
IF (IA.NE.F) GO TO 140	DSIN 250
WRITE(6,10900)	DSIN 255
READ(6,11000)NIN	DSIN 260
GO TO 160	DSIN 265
140 IF (IA.EQ.H) GO TO 150	DSIN 270
WRITE(6,10200)	DSIN 275
GO TO 130	DSIN 280
150 NIN=6	DSIN 285
160 CONTINUE	DSIN 290
C INPUT NUMBER OF POINTS	DSIN 295
170 WRITE(6,11100)	DSIN 300
WRITE(6,11200)	DSIN 305
READ(6,11300)NP	DSIN 310
IF (NP.LE.300) GO TO 180	DSIN 315
WRITE(6,11400)	DSIN 320
GO TO 170	DSIN 325
180 IF (NIN.NE.6) GO TO 190	DSIN 330
WRITE(6,11500)	DSIN 335
READ(NIN,11600)(DATA(I),I=1,NP)	DSIN 340
GO TO 210	DSIN 345

190 IF (NP.NE.0) READ(NIN,11600)(DATA(I),I=1,NP)	DSIN 350
IF (NP.NE.0) GO TO 210	DSIN 355
DO 200 I=1,300	DSIN 360
READ(NIN,11600,END=210)DATA(I)	DSIN 365
NP=I	DSIN 370
200 CONTINUE	DSIN 375
210 REWIND NIN	DSIN 380
WRITE(6,11700)	DSIN 385
WRITE(6,11800)NP	DSIN 390
RETURN	DSIN 395
10000 FORMAT(2X,'LINEAR OR LOG-NORMAL ANALYSIS? (3 CHAR)')	DSIN 400
10100 FORMAT(A5)	DSIN 405
10200 FORMAT(2X,'INVALID INPUT: TRY AGAIN.')	DSIN 410
10300 FORMAT(2X,'PERFORM KSL TEST? (YES OR NO)')	DSIN 415
10400 FORMAT(2X,'AND HOW ABOUT A CHI SQUARED TEST? (YES OR NO)')	DSIN 420
10500 FORMAT(2X,'INPUT TITLE OF PLOT,UP TO 19 CHAR.')	DSIN 425
10600 FORMAT(4A5)	DSIN 430
10700 FORMAT(2X,'EXP. DATA INPUT:HAND INPUT OR FILE? (4 CHAR)')	DSIN 435
10800 FORMAT(A4)	DSIN 440
10900 FORMAT(2X,'ENTER INPUT FILE NUMBER')	DSIN 445
11000 FORMAT(I2)	DSIN 450
11100 FORMAT(2X,'ENTER NUMBER OF DATA POINTS TO BE USED.')	DSIN 455
11200 FORMAT(2X,'(IF NUMBER = 0, ALL POINTS USED FOR FILE INPUT)')	DSIN 460
11300 FORMAT(I3)	DSIN 465
11400 FORMAT(2X,'NUMBER OF POINTS CANNOT EXCEED 300')	DSIN 470
11500 FORMAT(2X,'ENTER DATA, ONE VALUE PER LINE')	DSIN 475
11600 FORMAT(F10.0)	DSIN 480
11700 FORMAT(2X,'DATA INPUT COMPLETE: ANALYSIS STARTED.')	DSIN 485
11800 FORMAT(2X,'NUMBER OF POINTS =',I5)	DSIN 490
END	DSIN 495

SUBROUTINE DSNORM(X,P)	DSNO 0
CALLED BY DATA STATISTICS PKG	DSNO 5
C D.E. FIELDS, C.A. LITTLE, AND D.L. SHAEFFER	DSNO 10
C CODE, TERPED, A MODIFICATION BY D.E. FIELDS	DSNO 15
COMMON AP(2,101),U,S	DSNO 20
X1=(X-U)/S	DSNO 25
P=.5*(1.-FRF(-X1*.7071068))	DSNO 30
RETURN	DSNO 35
END	DSNO 40

SUBROUTINE DSORD(LIST,NOL2,NPTP)	DSOR 0
C DSORTM - MULTIPLE BYTE, LINEAR SELECTION, POINTER SORT	DSOR 5
C PROGRAM AUTHOR A. A. BROOKS	DSOR 10
C COMPUTING TECHNOLOGY CENTER, UNION CARBIDE CORP., NUCLEAR DIV..	DSOR 15

```

C OAK RIDGE, TENN.
C GREATLY REVISED
C BY D.E. FIELDS, C.A. LITTLE, AND D.L. SHAEFFER
CODE, TERPED, A MODIFICATION BY D.E. FIELDS
C CALLED BY DATA STATISTICS PKG
C NUMBER OF COMPARISONS=N**2/2
C LIST=DOUBLY DIMENSIONED ARRAY TO BE SORTED
C NDL2=HIGHEST INDEX OF SECOND DIMENSION OF LIST
C NPTR=POINTER ARRAY, DIMENSION GE NDL2
C LPTR=WORKING ARRAY, DIMENSION GE NDL2
COMMENT NO CHECK ON INITIAL ORDER
C *****
  DIMENSION NPTR(NDL2)
  REAL*4 LIST(NDL2)
COMMENT SET UP POINTER ARRAY
  DO 10 I=1,NDL2
10   NPTR(I)=I
C *****
  IF (NDL2.LE.1) RETURN
  IUP=NDL2-1
  DO 40 I=1,IUP
    JHOLD=I
    JLOW=I+1
    DO 30 J=JLOW,NDL2
      IF (LIST(NPTR(J)).GT.LIST(NPTR(JHOLD))) GO TO 30
20     JHOLD=J
30     CONTINUE
    NPTRH=NPTR(I)
    NPTR(I)=NPTR(JHOLD)
    NPTR(JHOLD)=NPTRH
40   CONTINUE
  RETURN
END

```

```

DSOR 20
DSOR 25
DSOR 30
DSOR 35
DSOR 40
DSOR 45
DSOR 50
DSOR 55
DSOR 60
DSOR 65
DSOR 70
DSOR 75
DSOR 80
DSOR 85
DSOR 90
DSOR 95
DSOR 100
DSOR 105
DSOR 110
DSOR 115
DSOR 120
DSOR 125
DSOR 130
DSOR 135
DSOR 140
DSOR 145
DSOR 150
DSOR 155
DSOR 160
DSOR 165
DSOR 170
DSOR 175
DSOR 180

```

```

SUBROUTINE DSPL0T(R,NP,PARM,IT,IV,TITL,COP,IPL)
C CALLED BY DATA STATISTICS PKG.
C D.E. FIELDS, C.A. LITTLE, AND D.L. SHAEFFER
CODE, TERPED, A MODIFICATION BY D.E. FIELDS
COMMON AP(2,101)
DIMENSION B(2),PARM(2,300),TITL(5),X(300),Y(300),LXAPAY(13)
DIMENSION IPAK(56),Y1(300),XLB(5)
DATA (LXAPAY(I),I=1,13)/'.14','.62','.23','.67','.16','.31','.50',
1 '.69','.84','.93','.97.7','.99.4','.99.9',MES/'R= S'/,DOL/'S'
CALL SETDEV(1,1)
DO 10 J=1,5
  XLB(J)=DOL
10 CONTINUE
WRITE(6,10000)
READ(6,10100)(XLB(J),J=1,5)
WRITE(6,10200)
IF (IV.NE.1) GO TO 20
WRITE(6,10300)
READ(6,10400)YOR
WRITE(6,10500)

```

```

DSPL 0
DSPL 5
DSPL 10
DSPL 15
DSPL 20
DSPL 25
DSPL 30
DSPL 35
DSPL 40
DSPL 45
DSPL 50
DSPL 55
DSPL 60
DSPL 65
DSPL 70
DSPL 75
DSPL 80
DSPL 85
DSPL 90
DSPL 95

```

```

      READ(6,10400)YST
      GO TO 30
20  WRITE(6,10600)
      READ(6,10400)YOR
      WRITE(6,10700)
      READ(6,10400)YCY
30  CONTINUE
      CALL BGNPL(-IPL)
      CALL SCMLX
      CALL PAGE(8.,9.5)
      CALL YTICKS(5)
      CALL TITLE(TITL,-100,'FRACT. PERTUBATIONS',0, XL9,100, 8., 5.)
      IF (IV.EQ.1) CALL GRAPH(-3.,.75,YOR,YST)
      IF (IV.EQ.2) CALL YLOG(-3.,.75,YOR,YCY)
C    PLOT DATA POINTS
      DO 40 N=1,NP
        Y(N)=PARM(1,N)
        IF (IT.EQ.2) Y(N)=EXP(Y(N))
        X(N)=PARM(2,N)
        IF (X(N).GT.3.) X(N)=3.
        IF (X(N).LT.-3.) X(N)=-3.
40    CONTINUE
      CALL CURVE(X,Y,NP,-1)
C    PLOT LS FIT LINE
      DO 50 N=1,NP
        Y1(N)=8(1)*PARM(2,N)+9(2)
        IF (IT.EQ.2) Y1(N)=EXP(Y1(N))
        X(N)=PARM(2,N)
        IF (X(N).GT.3.) X(N)=3.
        IF (X(N).LT.-3.) X(N)=-3.
50    CONTINUE
      CALL MARKER(3)
      CALL CURVE(X,Y1,NP,0)
C    TEMP PRINT OF LINEARIZED DATA*****
      DO 60 N=1,NP
60    CONTINUE
C    ***** FOR SPECIAL PLOT...*
      GO TO 70
      CALL LINES('INPUT DATA',IPAK,1)
      CALL LINES('LS FITS',IPAK,2)
      CALL LEGEND(IPAK,2,1.,4.)
70  CALL XLBAXS(LXARAY,1,13,8.,'CUMULATIVE PROBABILITY (%)', 100.0.,
1  0.)
      GO TO 80
      CALL MESSAG(MES,100,1.1,3.65)
      CALL REALNO(COR,3,1.4,3.65)
80  CALL ENDPL(IPL)
      RETURN
10000 FORMAT(2X,'ENTER VERTICAL AXIS LABEL')
10100 FORMAT(5A5)
10200 FORMAT(2X,'DEFINED ORDINATE LENGTH IS 5 INCHES...')
10300 FORMAT(2X,'ENTER Y ORIGIN.')
10400 FORMAT(F10.0)
10500 FORMAT(2X,'ENTER Y AXIS STEP SIZE (UNITS/INCH).')
10600 FORMAT(2X,'ENTER Y ORIGIN (NOT ZERO).')
10700 FORMAT(2X,'ENTER NUMBER OF INCHES PER LOG CYCLE.')
      END

```

```

DSPL 100
DSPL 105
DSPL 110
DSPL 115
DSPL 120
DSPL 125
DSPL 130
DSPL 135
DSPL 140
DSPL 145
DSPL 150
DSPL 155
DSPL 160
DSPL 165
DSPL 170
DSPL 175
DSPL 180
DSPL 185
DSPL 190
DSPL 195
DSPL 200
DSPL 205
DSPL 210
DSPL 215
DSPL 220
DSPL 225
DSPL 230
DSPL 235
DSPL 240
DSPL 245
DSPL 250
DSPL 255
DSPL 260
DSPL 265
DSPL 270
DSPL 275
DSPL 280
DSPL 285
DSPL 290
DSPL 295
DSPL 300
DSPL 305
DSPL 310
DSPL 315
DSPL 320
DSPL 325
DSPL 330
DSPL 335
DSPL 340
DSPL 345
DSPL 350
DSPL 355
DSPL 360
DSPL 365
DSPL 370
DSPL 375
DSPL 380

```

```

SUBROUTINE DSPROB(DATA, ITYPE, NP, IORD, PARM)
C   CALLED BY DATA STATISTICS PACKAGE
C   D.E. FIELDS, C.A. LITTLE, AND D.L. SHAEFFER
CODE, TERPED, A MODIFICATION BY D.E. FIELDS
COMMON AP(2,101)
DIMENSION DATA(300), IORD(300), PARM(2,300), X(5), Y(5), BPAR(4), C(4),
1 3), P(300), A(1), PR(1)
DATA MES/'YES'/
C   IORD SPECIFIES RANK ORDERING FOR DATA
C   PARM CONTAINS DATA ELEMENTS AND A, THE PANK ORDER
C   SPLINE FIT INITIALIZATION
DO 10 N=1,4
10  BPAR(N)=0.0
NX=5
IC=4
IER=0
M=1
C   DEFINE ELEMENTS OF PARM
FNP=NP
DO 50 J=1, NP
JC=IORD(J)
PARM(1,J)=DATA(JC)
C
C   DETERMINE 5 ADJOINING POINTS FOR SPLINE FIT
C   FJO=JO
C   COMPUTE CUMULATIVE PROBABILITY
PR(1)=(FJO-.375)/(FNP+.25)
IF (PR(1).GE..9999) PR(1)=.9999
DO 20 N=1,101
IF (PR(1).LT.AP(2,N)) GO TO 30
20  CONTINUE
30  NL=N-3
IF (NL.LT.0) NL=0
DO 40 N=1,5
NLPN=NL+N
X(N)=AP(2,NLPN)
Y(N)=AP(1,NLPN)
40  CONTINUE
CALL ICSCU(X,Y,NX,BPAR,C,IC,IER)
IF (IER.GE.129) WRITE(6,10000)JO,IER
CALL ICSEV(X,Y,NX,C,IC,PR,A,M,IER)
IF (IER.GE.33) WRITE(6,10100)JO,IER
P(JO)=PR(1)
PARM(2,JO)=A(1)
50  CONTINUE
WRITE(6,10200)
READ(6,10300)NES
IF (NES.NE.MES) GO TO 60
WRITE(6,10400)
DO 60 JO=1, NP
WRITE(6,10500)JO,PARM(1,JO),P(JO)
60  CONTINUE
DO 70 JO=1, NP
DATA(JO)=PARM(1,JO)
70  CONTINUE
RETURN
10000 FORMAT(2X,'ERROR FROM ICSCU, FROM DSPROB: DATA POINT', I2,

```

```

DSPR 0
DSPR 5
DSPR 10
DSPR 15
DSPR 20
DSPR 25
DSPR 30
DSPR 35
DSPR 40
DSPR 45
DSPR 50
DSPR 55
DSPR 60
DSPR 65
DSPR 70
DSPR 75
DSPR 80
DSPR 85
DSPR 90
DSPR 95
DSPR 100
DSPR 105
DSPR 110
DSPR 115
DSPR 120
DSPR 125
DSPR 130
DSPR 135
DSPR 140
DSPR 145
DSPR 150
DSPR 155
DSPR 160
DSPR 165
DSPR 170
DSPR 175
DSPR 180
DSPR 185
DSPR 190
DSPR 195
DSPR 200
DSPR 205
DSPR 210
DSPR 215
DSPR 220
DSPR 225
DSPR 230
DSPR 235
DSPR 240
DSPR 245
DSPR 250
DSPR 255
DSPR 260
DSPR 265
DSPR 270
DSPR 275
DSPR 280
DSPR 285
DSPR 290

```

```

1 ' , ERR = ' , I3)
10100 FORMAT(2X, 'ERROR FROM ICSEVU, FROM DSPROB: DATA POINT', 12,
1 ' , ERR = ' , I3)
10200 FORMAT(2X, 'PRINT ORDERED DATA (YES OR NO?)')
10300 FORMAT(A3)
10400 FORMAT(4X, 'J', 5X, 'VALUE', 6X, 'PROB')
10500 FORMAT(2X, I3, E11.3, F10.4)
END

```

DSPR 295
 DSPR 300
 DSPR 305
 DSPR 310
 DSPR 315
 DSPR 320
 DSPR 325
 DSPR 330

```

REAL FUNCTION ERF(X)
C16 DOUBLE PRECISION FUNCTION ERF(X, IFAIL)
C MARK 4 RELEASE NAG COPYRIGHT 1974.
C EDITED BY JOYCE CLARKE OXFORD OEG NUCLEAR PHYSICS 11TH SEP 1976
C FORTRAN MACRO VERSION FDIA24.TEC
C MARK 4.5 REVISED
C ERF(X)
C INTEGER NCF, IFAIL, J
C REAL XLO, XHI, ZERO, ONE, TWO, THREE, TWELVE, HALF, X, X2, BJP2,
C 1 BJP1, BJ, SUM
C
C07 REAL C(11)
C07 DATA NCF/11/, XLO, XHI /-6.0, 8.0/
C07 A, C(1), C(2), C(3), C(4), C(5), C(6), C(7), C(8), C(9), C(10), C(11)
C07 A/6.597255E-1, -4.536615E-1, 1.603983E-1, -4.49081E-2, 9.6414E-3
C07 A, -1.4425E-3, 1.010E-4, 1.21E-5, -3.6E-6, 1.0E-7, 1.0E-7/
C
C REAL C(16)
C DATA NCF /16/, XLO, XHI /-6.0, 12.0/, C(1), C(2), C(3), C(4), C(5),
C 1 C(6), C(7), C(8), C(9), C(10), C(11), C(12), C(13), C(14), C(15)
C 1 C(16) /6.5972554951E-1, -4.5366152053780E-1, 1.6039834082156E-1, -
C 1 4.490812115E-2, 9.64142865E-3, -1.44250808E-3, 1.0099437E-4,
C 1 1.208502E-5, -3.62542E-6, 7.187E-8, 8.914E-8, -7.42E-9, -2.39E-9,
C 1 3.1E-10, 8.E-11, -1.E-11/
C
C14 REAL C(21)
C14 DATA NCF/21/, XLO, XHI /-6.0, 14.0/
C14 A, C(1), C(2), C(3), C(4), C(5), C(6), C(7), C(8), C(9), C(10), C(11), C(12)
C14 A, C(13), C(14), C(15), C(16), C(17), C(18), C(19), C(20), C(21)
C14 A/6.5972554950607E-1, -4.5366152053780E-1, 1.6039834082156E-1
C14 A, -4.490812115112E-2, 9.64142865070E-3, -1.44250807774E-3
C14 A, 1.0099436735E-4, 1.208502453E-5, -3.62542060E-6, 7.186811E-8
C14 A, 8.914236E-8, -7.42157E-9, -2.38934E-9, 3.0622E-10, 7.789E-11
C14 A, -1.097E-11, -3.09E-12, 3.5E-13, 1.4E-13, -1.0E-14, -1.0E-14/
C
C DATA ZERO, ONE, TWO, THREE, TWELVE, HALF /0.0, 1.0, 2.0, 3.0, 12.0,
C 1 0.5/
C
C IF (X.LE.XLO) GO TO 30
C IF (X.GE.XHI) GO TO 40
C X2 = TWO - TWELVE/(ABS(X)+THREE)
C
C SUMMATION
C BJP2 = ZERO
C BJP1 = C(NCF)

```

ERF 0
 ERF 5
 ERF 10
 ERF 15
 ERF 20
 ERF 25
 ERF 30
 ERF 35
 ERF 40
 ERF 45
 ERF 50
 ERF 55
 ERF 60
 ERF 65
 ERF 70
 ERF 75
 ERF 80
 ERF 85
 ERF 90
 ERF 95
 ERF 100
 ERF 105
 ERF 110
 ERF 115
 ERF 120
 ERF 125
 ERF 130
 ERF 135
 ERF 140
 ERF 145
 ERF 150
 ERF 155
 ERF 160
 ERF 165
 ERF 170
 ERF 175
 ERF 180
 ERF 185
 ERF 190
 ERF 195
 ERF 200
 ERF 205
 ERF 210
 ERF 215
 ERF 220

J = NCF - 1	ERF	225
10 BJ = X2*BJP1 - BJP2 + C(J)	ERF	230
IF (J.EQ.1) GO TO 20	ERF	235
BJP2 = BJP1	ERF	240
BJP1 = BJ	ERF	245
J = J - 1	ERF	250
GO TO 10	ERF	255
20 SUM = HALF*(BJ-BJP2)*EXP(-X*X)	ERF	260
IF (X.LT.ZERO) GO TO 50	ERF	265
ERFC = SUM	ERF	270
GO TO 60	ERF	275
30 ERFC = TWO	ERF	280
GO TO 60	ERF	285
40 ERFC = ZERO	ERF	290
GO TO 60	ERF	295
50 ERFC = TWO - SUM	ERF	300
60 ERF=1.-ERFC	ERF	305
RETURN	ERF	310
END	ERF	315

SUBROUTINE GFIT (PDF,K,OBS,N,CELLS,COMP,CS,IDF,Q,IER)		GFIT	0
C		GFIT	5
C	-----S-----LIBRARY 1-----	GFIT	10
C		GFIT	15
C	FUNCTION - CHI-SQUARED GOODNESS OF FIT TEST	GFIT	20
C	USAGE - CALL GFIT(PDF,K,OBS,N,CELLS,COMP,CS,IDF,Q,IER)	GFIT	25
C	PARAMETERS PDF - THEORETICAL PROBABILITY DISTRIBUTION FUNCTION	GFIT	30
C	AGAINST WHICH SAMPLE IS TESTED. PDF IS A	GFIT	35
C	USER-SUPPLIED EXTERNAL FUNCTION.	GFIT	40
C	K - INPUT NUMBER OF EQUIPROBABLE CELLS INTO WHICH	GFIT	45
C	THE OBSERVATIONS ARE TO BE TALLIED.	GFIT	50
C	OBS - INPUT VECTOR OF LENGTH N, CONTAINING THE	GFIT	55
C	SAMPLE.	GFIT	60
C	N - LENGTH OF THE VECTOR OBS. (INPUT)	GFIT	65
C	CELLS - OUTPUT VECTOR OF LENGTH K CONTAINING COUNTS	GFIT	70
C	OF OBSERVATIONS WHICH FALL INTO THE	GFIT	75
C	EQUIPROBABLE CATEGORIES.	GFIT	80
C	COMP - OUTPUT VECTOR OF LENGTH K CONTAINING THE	GFIT	85
C	COMPONENTS OF THE CHI-SQUARED STATISTIC,	GFIT	90
C	K(CELLS-N/K)**2/N.	GFIT	95
C	CS - RESULTANT CHI-SQUARED STATISTIC.	GFIT	100
C	IDF - ON ENTRY CONTAINS THE NUMBER OF HYPOTHESIZED	GFIT	105
C	DISTRIBUTION FUNCTION PARAMETERS ESTIMATED	GFIT	110
C	FROM THE DATA, OBS. IT SHOULD BE ZERO	GFIT	115
C	IF NONE WERE ESTIMATED. ON EXIT, IT	GFIT	120
C	CONTAINS THE DEGREES OF FREEDOM OF THE	GFIT	125
C	STATISTIC CS.	GFIT	130
C	Q - OUTPUT PROBABILITY OF THE CHI-SQUARED	GFIT	135
C	STATISTIC EXCEEDING CS IF THE NULL	GFIT	140
C	HYPOTHESIS IS TRUE.	GFIT	145
C	IER - ERROR PARAMETER.	GFIT	150
C	TERMINAL ERROR = 128*N	GFIT	155
C	N=1 MEANS AN ERROR OCCURRED IN MDCDF1.	GFIT	160
C	N=2 MEANS K LESS THAN 2.	GFIT	165

```

C          WARNING ERROR = 32*N          GFIT 170
C          N=3 MEANS EXPECTED CELL VALUE IS LESS THAN 5GFIT 175
C          N=4 MEANS EXPECTED CELL VALUE IS LESS THAN 1GFIT 180
C          IN THIS CASE, Q IS SET TO NEGATIVE          GFIT 185
C          MACHINE INFINITY.          GFIT 190
C  PRECISION          - SINGLE          GFIT 195
C  REQD. IMSL ROUTINES - MDCDFI,UERTST          GFIT 200
C  LANGUAGE          - FORTRAN          GFIT 205
C-----GFIT 210
C  LATEST REVISION    - APRIL 3, 1975          GFIT 215
C          GFIT 220
C          GFIT 225
C          DIMENSION OBS(N),CELLS(K),COMP(K)          GFIT 230
C          DOUBLE PRECISION CS1,Q1          GFIT 235
C          DATA RINFM/-1.E30/          GFIT 240
C          WRITE(6,10000)          GFIT 245
C          IER = 0          GFIT 250
C          IF (K.LT.2) GO TO 60          GFIT 255
C          GFIT 260
C          DO 10 I=1,K          INITIALIZE COUNTS          GFIT 265
C          10 CELLS(I) = 0.0          GFIT 270
C          GFIT 275
C          DO 20 I=1,N          GENERATE COUNTS          GFIT 280
C          X = OBS(I)          GFIT 285
C          CALL PDF(X,P)          GFIT 290
C          J = K*P + 1          GFIT 295
C          IF (P.EQ.1.0) J = K          GFIT 300
C          20 CELLS(J) = CELLS(J) + 1.0          GFIT 305
C          GFIT 310
C          DETERMINE EXPECTED VALUES OF CELLS          GFIT 315
C          EN = N          GFIT 320
C          EK = K          GFIT 325
C          E = EN/EK          GFIT 330
C          DO 30 I=1,K          GFIT 335
C          CELL1 = CELLS(I)-E          GFIT 340
C          GFIT 345
C          FIND COMPONENTS OF STATISTICS          GFIT 350
C          30 COMP(I) = EK*(CELL1*CELL1)/EN          GFIT 355
C          CS1 = 0.000          GFIT 360
C          DO 40 I=1,K          GFIT 365
C          40 CS1 = CS1 + COMP(I)          GFIT 370
C          CS = CS1          GFIT 375
C          IDF = K-1-IDF          GFIT 380
C          IF (E.LT.5.0) IER = 35          GFIT 385
C          IF (E.LT.1.0) GO TO 50          GFIT 390
C          GFIT 395
C          FIND PROBABILITY          GFIT 400
C          CALL MDCDFI(CS1,IDF,Q1,IET)          GFIT 405
C          IF (IET.GT.127) GO TO 70          GFIT 410
C          Q = Q1          GFIT 415
C          IF (IER-32) 90,80,80          GFIT 420
C          50 IER = 36          GFIT 425
C          Q = RINFM          GFIT 430
C          GO TO 80          GFIT 435
C          60 IER = 130          GFIT 440
C          GO TO 80          GFIT 445
C          70 IER = IET          GFIT 450
C          80 CONTINUE          GFIT 455
C          CALL UERTST(IER,6HGFIT)          GFIT 460
C          90 RETURN          GFIT 465
C          10000 FORMAT(2X,'BEGIN GFIT')          GFIT 470
C          END          GFIT 475

```

```

SUBROUTINE ICSEVU (X,Y,NX,C,IC,U,S,M,IER)
C-----LIBRARY 1-----
C
C FUNCTION      - EVALUATION OF A CUBIC SPLINE.
C USAGE        - CALL ICSEVU(X,Y,NX,C,IC,U,S,M,IER)
C PARAMETERS   X  - VECTOR OF LENGTH NX CONTAINING THE ABCISSAE
C                 OF THE NX DATA POINTS (X(I),Y(I)) I=1,...,
C                 NX (INPUT). X MUST BE ORDERED SO THAT
C                 X(I) .LT. X(I+1).
C                 Y  - VECTOR OF LENGTH NX CONTAINING THE ORDINATES
C                 (OR FUNCTION VALUES) OF THE NX DATA POINTS
C                 (INPUT).
C                 NX - NUMBER OF ELEMENTS IN X AND Y (INPUT).
C                 NX MUST BE .GE. 2.
C                 C  - SPLINE COEFFICIENTS (INPUT). C IS AN NX-1 BY
C                 3 MATRIX.
C                 IC - ROW DIMENSION OF MATRIX C IN THE CALLING
C                 PROGRAM (INPUT). IC MUST BE .GE. NX-1.
C                 U  - VECTOR OF LENGTH M CONTAINING THE ABCISSAE
C                 OF THE M POINTS AT WHICH THE CUBIC SPLINE
C                 IS TO BE EVALUATED (INPUT).
C                 S  - VECTOR OF LENGTH M (OUTPUT).
C                 THE VALUE OF THE SPLINE APPROXIMATION AT
C                 U(I) IS
C                 S(I) = ((C(J,3)*D+C(J,2))*D+C(J,1))*D+Y(J)
C                 WHERE X(J) .LE. U(I) .LT. X(J+1) AND
C                 D = U(I)-X(J).
C                 M  - NUMBER OF ELEMENTS IN U AND S (INPUT).
C                 IER - ERROR PARAMETER.
C                 WARNING ERROR
C                 IER = 33, U(I) IS LESS THAN X(1).
C                 IER = 34, U(I) IS GREATER THAN X(NX).
C PRECISION    - SINGLE/DOUBLE
C REQD. IMSL ROUTINES - UERTST
C LANGUAGE     - FORTRAN
C-----
C LATEST REVISION - AUGUST 9, 1974
C
C DIMENSION X(NX),Y(NX),C(IC,3),U(M),S(M)
C DOUBLE PRECISION C,D,DD,S,U,X,Y,ZERO
C I=1
C ZERO=0.
C
C             INITIALIZE ERROR PARAMETERS
C JER = 0
C KER = 0
C IF (M.LE.0) GO TO 100
C NXM1 = NX-1
C IF (1.GT.NXM1) I = 1
C
C             EVALUATE SPLINE AT M POINTS
C DO 80 K=1,M
C
C             FIND THE PROPER INTERVAL
C D = U(K)-X(I)
C IF (D) 10,50,30
C 10 IF (1.E0.1) GO TO 60
C I = I-1

```

```

ICSE 0
ICSE 5
ICSE 10
ICSE 15
ICSE 20
ICSE 25
ICSE 30
ICSE 35
ICSE 40
ICSE 45
ICSE 50
ICSE 55
ICSE 60
ICSE 65
ICSE 70
ICSE 75
ICSE 80
ICSE 85
ICSE 90
ICSE 95
ICSE 100
ICSE 105
ICSE 110
ICSE 115
ICSE 120
ICSE 125
ICSE 130
ICSE 135
ICSE 140
ICSE 145
ICSE 150
ICSE 155
ICSE 160
ICSE 165
ICSE 170
ICSE 175
ICSE 180
ICSE 185
ICSE 190
ICSE 195
ICSE 200
ICSE 205
ICSE 210
ICSE 215
ICSE 220
ICSE 225
ICSE 230
ICSE 235
ICSE 240
ICSE 245
ICSE 250
ICSE 255
ICSE 260
ICSE 265
ICSE 270
ICSE 275
ICSE 280

```


	D = U(K)-X(I)	ICSE 285
	IF (D) 10,50,40	ICSE 290
20	I = I+1	ICSE 295
	D = DD	ICSE 300
30	IF (I.GE.NX) GO TO 70	ICSE 305
	DD = U(K)-X(I+1)	ICSE 310
	IF (DD.GE.ZERO) GO TO 20	ICSE 315
	IF (D.EQ.ZERO) GO TO 50	ICSE 320
	PERFORM EVALUATION	ICSE 325
40	S(K) = ((C(I,3)*D+C(I,2))*D+C(I,1))*D+Y(I)	ICSE 330
	GO TO 80	ICSE 335
50	S(K) = Y(I)	ICSE 340
	GO TO 80	ICSE 345
	WARNING - U(I) .LT. X(1)	ICSE 350
60	JER = 33	ICSE 355
	GO TO 40	ICSE 360
	IF U(I) .GT. X(NX) - WARNING	ICSE 365
70	IF (DD.GT.ZERO) KER = 34	ICSE 370
	D = U(K)-X(NXM1)	ICSE 375
	I = NXM1	ICSE 380
	GO TO 40	ICSE 385
80	CONTINUE	ICSE 390
	IER = MAXO(JER,KER)	ICSE 395
90	CONTINUE	ICSE 400
	IF (JER.GT.0) CALL UERTST(JER,6HICSEVU)	ICSE 405
	IF (KER.GT.0) CALL UERTST(KER,6HICSEVU)	ICSE 410
100	RETURN	ICSE 415
	END	ICSE 420

SUBROUTINE ICSICU (X,Y,NX,BPAR,C,IC,IER)		ICSI 0
		ICSI 5
C-ICSICU-----D-----LIBRARY 1-----		ICSI 10
C		ICSI 15
C	FUNCTION - INTERPOLATORY APPROXIMATION BY CUBIC SPLINES	ICSI 20
C	WITH ARBITRARY SECOND DERIVATIVE END	ICSI 25
C	CONDITIONS.	ICSI 30
C	USAGE - CALL ICSICU(X,Y,NX,BPAR,C,IC,IER)	ICSI 35
C	PARAMETERS X - VECTOR OF LENGTH NX CONTAINING THE ABSCISSAE	ICSI 40
C	OF THE NX DATA POINTS (X(I),Y(I)) I=1,....	ICSI 45
C	NX (INPUT). X MUST BE ORDERED SO THAT	ICSI 50
C	X(I) .LT. X(I+1).	ICSI 55
C	Y - VECTOR OF LENGTH NX CONTAINING THE ORDINATES	ICSI 60
C	(OR FUNCTION VALUES) OF THE NX DATA POINTS	ICSI 65
C	(INPUT).	ICSI 70
C	NX - NUMBER OF ELEMENTS IN X AND Y (INPUT). NX	ICSI 75
C	MUST BE .GE. 2.	ICSI 80
C	BPAR - VECTOR OF LENGTH 4 CONTAINING THE END	ICSI 85
C	CONDITION PARAMETERS (INPUT).	ICSI 90
C	2.0*SPP(1)+BPAR(1)*SPP(2) = BPAR(2),	ICSI 95
C	BPAR(3)*SPP(NX-1)+2.0*SPP(NX) = BPAR(4),	ICSI 100
C	WHERE SPP(I) = SECOND DERIVATIVE OF THE	ICSI 105
C	CUBIC SPLINE FUNCTION S EVALUATED AT X(I).	ICSI 110
C	C - SPLINE COEFFICIENTS (OUTPUT). C IS AN NX-1 BY	ICSI 115
C	3 MATRIX. THE VALUE OF THE SPLINE	ICSI 120

```

C          APPROXIMATION AT T IS ICSI 125
C          S(T) = ((C(I,3)*D+C(I,2))*D+C(I,1))*D+Y(I) ICSI 130
C          WHERE X(I) .LE. T .LT. X(I+1) AND ICSI 135
C          D = T-X(I). ICSI 140
C          IC - ROW DIMENSION OF MATRIX C IN THE CALLING ICSI 145
C          PROGRAM (INPUT). IC MUST BE .GE. NX-1. ICSI 150
C          IER - ERROR PARAMETER. ICSI 155
C          TERMINAL ERROR ICSI 160
C          IER = 129, IC IS LESS THAN NX-1. ICSI 165
C          IER = 130, NX IS LESS THAN 2. ICSI 170
C          IER = 131, INPUT ABSCISSA ARE NOT ORDERED ICSI 175
C          SO THAT X(1) .LT. X(2) ... .LT. X(NX). ICSI 180
C          PRECISION - SINGLE/DOUBLE ICSI 185
C          REQD. INSL ROUTINES - UERTST ICSI 190
C          LANGUAGE - FORTRAN ICSI 195
C----- ICSI 200
C          LATEST REVISION - JULY 29, 1974 ICSI 205
C ICSI 210
C ICSI 215
C          DIMENSION X(NX),Y(NX),BPAP(4),C(IC,3) ICSI 220
C          DOUBLE PRECISION BPAP,C,DX,DXJ,DXJP1,DXP,DYJ,DYJP1,HALF,ONE,PJ, ICSI 225
C          SIX,SIXI,TWO,X,Y,YPPA,YPPB,ZERO ICSI 230
C          EQUIVALENCE (DXJ,YPPB),(PJ,SIXI),(DXJP1,YPPA) ICSI 235
C          ZERO=0.0 ICSI 240
C          HALF=.5 ICSI 245
C          ONE=1. ICSI 250
C          TWO=2. ICSI 255
C          SIX=6. ICSI 260
C          CHECK ERROR CONDITIONS ICSI 265
C          IER = 0 ICSI 270
C          NXMI = NX-1 ICSI 275
C          IF (IC.LT.NXMI) GO TO 60 ICSI 280
C          IF (NX.LT.2) GO TO 70 ICSI 285
C          IF (NX.EQ.2) GO TO 20 ICSI 290
C          COMPUTE COEFFICIENTS AND RIGHT ICSI 295
C          HAND SIDE OF THE TRIDIAGONAL ICSI 300
C          SYSTEM DEFINING THE SECOND ICSI 305
C          DERIVATIVES OF THE SPLINE ICSI 310
C          INTERPOLANT FOR (X,Y) ICSI 315
C          C(J,1) = LAMBDA(J) ICSI 320
C          C(J,2) = MU(J) ICSI 325
C          C(J,3) = D(J) ICSI 330
C          DXJ = X(2)-X(1) ICSI 335
C          IF (DXJ.LE.ZERO) GO TO 80 ICSI 340
C          DYJ = Y(2)-Y(1) ICSI 345
C          DO 10 J=2,NXMI ICSI 350
C          DXJP1 = X(J+1)-X(J) ICSI 355
C          IF (DXJP1.LE.ZERO) GO TO 80 ICSI 360
C          DYJP1 = Y(J+1)-Y(J) ICSI 365
C          DXP = DXJ+DXJP1 ICSI 370
C          C(J,1) = DXJP1/DXP ICSI 375
C          C(J,2) = ONE-C(J,1) ICSI 380
C          C(J,3) = SIX*(DYJP1/DXJP1-DYJ/DXJ)/DXP ICSI 385
C          DXJ = DXJP1 ICSI 390
C          DYJ = DYJP1 ICSI 395
C          10 CONTINUE ICSI 400
C          FACTOR THE TRIDIAGONAL MATRIX ICSI 405
C          AND SOLVE FOR U ICSI 410
C          C(J,2) = U(J) ICSI 415
C          C(J,1) = O(J) ICSI 420
C          BPAP(1) = LAMBDA(1) ICSI 425
C          BPAP(2) = D(1) ICSI 430
C          BPAP(3) = MU(NX) ICSI 435

```

	BPAR(4) = D(NX)	ICSI 440
20	C(1,1) = -BPAR(1)*HALF	ICSI 445
	C(1,2) = BPAR(2)*HALF	ICSI 450
	IF (NX.EQ.2) GO TO 40	ICSI 455
	DO 30 J=2,NXM1	ICSI 460
	PJ = C(J,2)*C(J-1,1)+TWO	ICSI 465
	C(J,1) = -C(J,1)/PJ	ICSI 470
	C(J,2) = (C(J,3)-C(J,2)*C(J-1,2))/PJ	ICSI 475
30	CONTINUE	ICSI 480
	SOLVE FOR CUBIC COEFFICIENTS	ICSI 485
	OF SPLINE INTERPOLANT	ICSI 490
	C(J,1), C(J,2), AND C(J,3)	ICSI 495
40	YPPB = (BPAR(4)-BPAR(3)*C(NXM1,2))/(BPAR(3)*C(NXM1,1)+TWO)	ICSI 500
	SIXI = ONE/SIX	ICSI 505
	DO 50 I=1,NXM1	ICSI 510
	J = NX-I	ICSI 515
	YPPA = C(J,1)*YPPB+C(J,2)	ICSI 520
	DX = X(J+1)-X(J)	ICSI 525
	C(J,3) = SIXI*(YPPB-YPPA)/DX	ICSI 530
	C(J,2) = HALF*YPPA	ICSI 535
	C(J,1) = (Y(J+1)-Y(J))/DX-(C(J,2)*C(J,3)+DX)*DX	ICSI 540
	YPPB = YPPA	ICSI 545
50	CONTINUE	ICSI 550
	GO TO 100	ICSI 555
60	IER = 129	ICSI 560
	GO TO 90	ICSI 565
70	IEP = 130	ICSI 570
	GO TO 90	ICSI 575
80	IER = 131	ICSI 580
90	CONTINUE	ICSI 585
	CALL UERTST(IEP,6HICSICU)	ICSI 590
100	RETURN	ICSI 595
	END	ICSI 600

	SUBROUTINE LINPLT(IPLT,YORG,YSTEP,PARM,ICOL,INP,B)	LINP 0
	DIMENSION IPLT(61,122),PARM(2,300),B(2)	LINP 5
	DATA KDATA/'D'//,KFIT/'F'//,KCOR/'*'/	LINP 10
	YNORM=12.*YSTEP	LINP 15
	XNORM=6.	LINP 20
	XORG=-3.	LINP 25
	DO 10 I=1,INP	LINP 30
	CALL NORM(PARM(1,I),YORG,60,YNORM,IROW)	LINP 35
	CALL NORM(PARM(2,I),XORG,ICOL,XNORM,ICOLUM)	LINP 40
	ICOLUM=ICOLUM+1	LINP 45
	IF (IROW.LT.1.OR.IROW.GT.61) GO TO 10	LINP 50
	IF (ICOLUM.LT.2.OR.ICOLUM.GT.122) GO TO 10	LINP 55
	IPLT(IROW,ICOLUM)=KDATA	LINP 60
10	CONTINUE	LINP 65
	DO 30 M=0,60	LINP 70
	YFV=((M*YNORM)/60.)+YORG	LINP 75
	XFV=(YFV-B(2))/B(1)	LINP 80
	CALL NORM(XFV,XORG,ICOL,XNORM,IFX)	LINP 85
	IFX=IFX+1	LINP 90
	IF (IFX.LT.2.OR.IFX.GT.122) GO TO 30	LINP 95

	IF (IPLOT(M+1, IFX).EQ.KDATA) GO TO 20	LINP 100
	IPLOT(M+1, IFX)=KFIT	LINP 105
	GO TO 30	LINP 110
20	IPLOT(M+1, IFX)=KCOR	LINP 115
30	CONTINUE	LINP 120
	RETURN	LINP 125
	END	LINP 130

	SUBROUTINE LOGPLY(IPLOT, YORG, ICYCLE, PARM, INP, B, ICOL)	LOGP 0
	DIMENSION IPLOT(61, 122), PARM(2, 300), B(2)	LOGP 5
	DATA KDATA/'D'//, KFIT/'F'//, KCOR/'*'/	LOGP 10
	XORG=-3.	LOGP 15
	XNORM=6.	LOGP 20
	IER=0	LOGP 25
	M=1	LOGP 30
	ITIER=60/ICYCLE	LOGP 35
	ZMIN=ALOG(YORG)	LOGP 40
	DO 70 L=1, ICYCLE	LOGP 45
	ZMAX=ALOG(YORG)+(ALOG(10.)*L)	LOGP 50
	ORFAC=ZMIN	LOGP 55
	IF (M.GT.INP) GO TO 40	LOGP 60
	DO 30 I=M, INP	LOGP 65
	IF (PARM(1, I).LT.ZMIN) GO TO 30	LOGP 70
	IF (PARM(1, I).GT.ZMAX) GO TO 40	LOGP 75
	ALNYV=PARM(1, I)	LOGP 80
	CALL NORM(ALNYV, ORFAC, ITIER, ALOG(10.), IROW)	LOGP 85
	IROW=IROW+(ITIER*(L-1))	LOGP 90
	IF (IROW.GE.1.AND.IROW.LE.61) GO TO 10	LOGP 95
	IER=IER+1	LOGP 100
	WRITE(6, 10000)	LOGP 105
	IF (IER.GT.2) RETURN	LOGP 110
	GO TO 30	LOGP 115
10	CALL NORM(PARM(2, I), XORG, ICOL, XNORM, ICOLUM)	LOGP 120
	ICOLUM=ICOLUM+1	LOGP 125
	IF (ICOLUM.GE.2.AND.ICOLUM.LE.122) GO TO 20	LOGP 130
	GO TO 30	LOGP 135
20	IPLOT(IROW, ICOLUM)=KDATA	LOGP 140
30	CONTINUE	LOGP 145
40	M=I	LOGP 150
	DO 60 K=1, ITIER	LOGP 155
	IFYV=K+(ITIER*(L-1))	LOGP 160
	YFV=((IFYV*ALOG(10.))/ITIER)+ALOG(YORG)	LOGP 165
	XFV=(YFV-B(2))/B(1)	LOGP 170
	CALL NORM(XFV, XORG, ICOL, XNORM, IXFV)	LOGP 175
	IXFV=IXFV+1	LOGP 180
	IF (IXFV.LT.2.OR.IXFV.GT.122) GO TO 60	LOGP 185
	IF (IPLOT(IFYV, IXFV).EQ.KDATA) GO TO 50	LOGP 190
	IPLOT(IFYV, IXFV)=KFIT	LOGP 195
	GO TO 60	LOGP 200
50	IPLOT(IFYV, IXFV)=KCOR	LOGP 205
60	CONTINUE	LOGP 210
	ZMIN=ZMAX	LOGP 215
70	CONTINUE	LOGP 220
	RETURN	LOGP 225

10000 FORMAT(/' ERROR IN LOG PLOT NORMALIZATION PARAMETERS'
END

LOGP 230
LOGP 235

```

SUBROUTINE MDGDFI (CS,N,Q,IER)
C
C-----LIBRARY 1-----
C
C FUNCTION          - CHI-SQUARED PROBABILITY DISTRIBUTION FUNCTION
C USAGE            - CALL MDGDFI(CS,N,Q,IER)
C PARAMETER      CS - INPUT VALUE FROM WHICH THE INTEGRATION IS
C                   N - INPUT DEGREES OF FREEDOM, NOT LESS THAN ONE.
C                   Q - PROBABILITY OF EXCEEDING CS.
C                   IER - ERROR PARAMETER.
C                   TERMINAL EPROR=128*N.
C                   N = 1 INDICATES THAT CS OR N(DF) IS OUT OF
C                   RANGE.
C PRECISION        - DOUBLE
C REQD. IMSL ROUTINES - UERTST
C LANGUAGE         - FORTRAN
C-----
C LATEST REVISION - JANUARY 1, 1977
C
C IMPLICIT REAL*8(A-H,O-Z)
C
C CHECK INPUT RANGES
C IF ((CS.LT.0.0).OR.(N.LT.1)) GO TO 80
C A =.500*CS
C IER=0
C Y=0.0
C IE=0
C
C SET EVEN DEGREES OF FREEDOM (N)
C INDICATOR.
C IF (N.EQ.(N/2)*2) IE=1
C IF (((IE.EQ.1).OR.(N.GT.2)).AND.(A.LE.170.000)) Y=DEXP(-A)
C S=Y
C
C IF N IS EVEN, SET S = 0 FOR DF = 2.
C IF N IS ODD, SET S = 0 FOR DF = 1.
C IF (IE.NE.0) GO TO 10
C DSQA=DSQRT(A)
C S=0.0
C IF (DSQA.LT.13.3) S = 1.-ERF(DSQA)
10 IF (N.GT.2) GO TO 30
20 Q=S
GO TO 100
C
C USE RECURRENCE FORMULA, REFERENCE 2.
30 XD=.500*(N-1)
Z=1.00
IF (IE.EQ.0) Z=.500
IF (A.LE.170.000) GO TO 50
E=0.000
IF (IE.EQ.0) E=0.572364942924698000
C=0LOG(A)
40 IF (Z.GT.XD) GO TO 20

```

E=DLOG(Z)+E	MDCD 255
ARG=C*Z-A-E	MDCD 260
IF (ARG.GE.-170.000) S=DEXP(ARG)+S	MDCD 265
Z=Z+1.00	MDCD 270
GO TO 40	MDCD 275
50 E=1.000	MDCD 280
C=0.000	MDCD 285
IF (A.EQ.0.0) GO TO 70	MDCD 290
IF (IE.EQ.0) E=0.56418958354775600/DSQA	MDCD 295
60 E=E*A/Z	MDCD 300
IF (C.EQ.C+E) GO TO 70	MDCD 305
C=C+E	MDCD 310
Z=Z+1.00	MDCD 315
IF (Z.LE.XD) GO TO 60	MDCD 320
70 Q=C*Y+S	MDCD 325
GO TO 100	MDCD 330
80 IE=129	MDCD 335
90 CONTINUE	MDCD 340
CALL UERTST(IE,'MDCDFI')	MDCD 345
100 RETURN	MDCD 350
END	MDCD 355

SUBROUTINE MOSMR (X,P1,P2)		MDSM 0
C		MDSM 5
C	-----S-----LIBRARY 1-----	MDSM 10
C	FUNCTION - KOLMOGOROV-SMIRNOV STATISTICS ASYMPTOTIC	MDSM 15
C	DISTRIBUTION FUNCTION VALUES	MDSM 20
C	USAGE - CALL MOSMR(X,P1,P2)	MDSM 25
C	PARAMETERS X - GIVEN CONSTANT TO WHICH INTEGRATION IS	MDSM 30
C	PERFORMED. PROBABILITY THAT A RANDOM	MDSM 35
C	VARIABLE Z IS LESS THAN OR EQUAL TO X IS	MDSM 40
C	CALCULATED, WHERE Z IS DISTRIBUTED SMIRNOV	MDSM 45
C	P1 - PROB (Z IS LESS THAN OR EQUAL TO X) FOR A	MDSM 50
C	ONE SIDED TEST	MDSM 55
C	P2 - PROB (Z IS LESS THAN OR EQUAL TO X) FOR A	MDSM 60
C	TWO SIDED TEST	MDSM 65
C	PRECISION - SINGLE	MDSM 70
C	LANGUAGE - FORTRAN	MDSM 75
C	-----	MDSM 80
C	LATEST REVISION - OCTOBER 11,1971	MDSM 85
C		MDSM 90
C	SUBROUTINE MOSMR (X,P1,P2)	MDSM 95
C		MDSM 100
C	X1 = -X*X	MDSM 105
C	P1 = EXP(X1*X1)	MDSM 110
C	IF (X.GE..22) GO TO 10	MDSM 115
C	P2 = 0.0	MDSM 120
C	GO TO 40	MDSM 125
C	10 X8 = 8.*X1	MDSM 130
C	IF (X.GT.0.8) GO TO 20	MDSM 135
C	P2 = (2.506628/X)*EXP(9.869604/X8)	MDSM 140
C	GO TO 40	MDSM 145
C	20 IF (X.LE.3.15) GO TO 30	MDSM 150
C		MDSM 155

P2 = 1.0	MDSM 160
P1 = 1.0	MDSM 165
GO TO 50	MDSM 170
30 X8 = P1*P1	MDSM 175
X8 = X8*X8	MDSM 180
P2 = P1*X8*X8-X8*P1	MDSM 185
P2 = 1.0-P2-P2	MDSM 190
40 P1 = 1.0-P1	MDSM 195
50 RETURN	MDSM 200
END	MDSM 205

SUBROUTINE NKS1 (PDF,X,N,PDIF,IER)		NKS1 0
C		NKS1 5
C	-----S-----LIBRARY 1-----	NKS1 10
C		NKS1 15
C	FUNCTION - KOLMOGOROV-SMIRNOV ONE-SAMPLE TEST	NKS1 20
C	USAGE - CALL NKS1(PDF,X,N,PDIF,IER)	NKS1 25
C	PARAMETERS PDF - USER SUPPLIED SUBROUTINE WHICH MUST DETERMINE	NKS1 30
C	THE ORDINATE OF THEORETICAL DISTRIBUTION	NKS1 35
C	FUNCTION AT X(I). (THE AREA OF THE	NKS1 40
C	THEORETICAL DENSITY FROM MINUS INFINITY TO	NKS1 45
C	X(I))	NKS1 50
C	X - SAMPLE - VECTOR OF N INDEPENDENT OBSERVATIONS.	NKS1 55
C	PRIOR TO ENTRY, X MUST BE SORTED INTO	NKS1 60
C	ASCENDING ORDER. THIS MIGHT BE ACCOM-	NKS1 65
C	PLISHED BY CALLING IMSL SUBROUTINE VSORTA.	NKS1 70
C	N - LENGTH OF VECTOR X (SAMPLE SIZE)	NKS1 75
C	PDIF - VECTOR OF LENGTH 6. PDIF(I) CONTAINS, WHEN	NKS1 80
C	I =----	NKS1 85
C	1--D, THE MAXIMUM OF D+,D- (BELOW)	NKS1 90
C	2--D+, THE SUPREMUM OF THE DIFFERENCES	NKS1 95
C	S - F (SAMPLE - THEORETICAL	NKS1 100
C	DISTRIBUTION FUNCTIONS)	NKS1 105
C	3--D-, THE SUPREMUM OF THE DIFFERENCES	NKS1 110
C	F - S.	NKS1 115
C	4--Z, THE STATISTIC USED TO OBTAIN THE	NKS1 120
C	PROBABILITIES.	NKS1 125
C	5--THE PROBABILITY OF THE STATISTIC	NKS1 130
C	EXCEEDING Z IF THE HYPOTHESIS OF	NKS1 135
C	EQUALITY IS TRUE AND THE ALTERNATIVE	NKS1 140
C	IS ONE SIDED.	NKS1 145
C	6--THE PROBABILITY OF EXCEEDING Z WHEN THE	NKS1 150
C	HYPOTHESIS OF EQUALITY IS TRUE AND THE	NKS1 155
C	ALTERNATIVE IS TWO SIDED.	NKS1 160
C	IER - ERROR PARAMETER. WARNING ERROR = 32+N.	NKS1 165
C	N = 1 --SAMPLE SIZE IS LESS THAN 80.	NKS1 170
C	PRECISION - SINGLE	NKS1 175
C	REQD. IMSL ROUTINES - MDSMP,UERTST	NKS1 180
C	LANGUAGE - FORTRAN	NKS1 185
C	-----	NKS1 190
C	LATEST REVISION - MAY 15, 1975	NKS1 195
C		NKS1 200
C	SUBROUTINE NKS1(PDF,X,N,PDIF,IER)	NKS1 205
C		NKS1 210

DIMENSION PDIF(6),X(N)	NKS1 215
IER = 0	NKS1 220
IF (N.GT.80) GO TO 10	NKS1 225
C SAMPLE SIZE WARNING	NKS1 230
IER = 33	NKS1 235
10 PDIF(2) = 0.	NKS1 240
PDIF(3) = 0.	NKS1 245
FN = N	NKS1 250
FN = 1./FN	NKS1 255
CALL PDF(X(1),F)	NKS1 260
S = 0.	NKS1 265
N1 = N - 1	NKS1 270
II = 1	NKS1 275
C CHECK FOR TIES	NKS1 280
20 DO 30 I=II,N1	NKS1 285
J = I	NKS1 290
IF (X(J).LT.X(J+1)) GO TO 50	NKS1 295
30 CONTINUE	NKS1 300
40 J = N	NKS1 305
50 II = J + 1	NKS1 310
C RETAIN SAMPLE PDF VALUES	NKS1 315
C EVALUATE SAMPLE PDF	NKS1 320
SS = FLOAT(J)*FN	NKS1 325
C EVALUATE THEORETICAL PDF	NKS1 330
E = F-S	NKS1 335
EE = SS-F	NKS1 340
IF (J.EQ.N) GO TO 60	NKS1 345
CALL PDF(X(J+1),F)	NKS1 350
S=SS	NKS1 355
C RETAIN MAX, MIN VALUES	NKS1 360
60 PDIF(2) = AMAX1(PDIF(2),EE)	NKS1 365
PDIF(3) = AMAX1(PDIF(3),E)	NKS1 370
C CHECK FOR LAST SAMPLE POINT	NKS1 375
IF (II-N) 20,40,70	NKS1 380
70 PDIF(1) = AMAX1(PDIF(2),PDIF(3))	NKS1 385
PDIF(4) = PDIF(1)/SQRT(FN)	NKS1 390
CALL MDSP(PDIF(4),E,S)	NKS1 395
PDIF(5) = 1. - E	NKS1 400
PDIF(6) = 1. - S	NKS1 405
IF (IER.EQ.0) GO TO 90	NKS1 410
C CHECK ERROR INDICATOR	NKS1 415
80 CONTINUE	NKS1 420
CALL UERTST(IER,6H NKS1)	NKS1 425
90 RETURN	NKS1 430
END	NKS1 435

SUBROUTINE NORM(DATA,ORIGIN,ISIZE,ANORM,IPOS)	NORM 0
DATNO=DATA-ORIGIN	NORM 5
PRCNOR=DATNO*ISIZE	NORM 10
AROW=PRCNOR/ANORM	NORM 15
IPOS=INT(AROW*.555)+1	NORM 20
RETURN	NORM 25
END	NORM 30

SUBROUTINE PRNCON(PARM,B,INP,ITYPE,TITLE)	PRNC 0
DIMENSION IPLOT(61,122),KYLAB(4),PARM(2,300),B(2),TITLE(4)	PRNC 5
DATA KSTAR/'*'/,KGRID/'.'/,KTIC/'+'/,KBLK/'.'/	PRNC 10
WRITE(6,10000)	PRNC 15
READ(6,10100)(KYLAB(I),I=1,4)	PRNC 20
10 WRITE(6,10200)	PRNC 25
READ(6,10300)ICOL	PRNC 30
IF (ICOL.NE.120.AND.ICOL.NE.60) GO TO 10	PRNC 35
20 WRITE(6,10400)	PRNC 40
READ(6,10500)ID	PRNC 45
IF (ID.NE.'TTY') GO TO 30	PRNC 50
IDEV=6	PRNC 55
GO TO 60	PRNC 60
30 IF (ID.NE.'DSK') GO TO 50	PRNC 65
40 WRITE(6,11400)	PRNC 70
READ(6,11500)IDISK	PRNC 75
IF (IDISK.LT.7.OR.IDISK.GT.63) GO TO 40	PRNC 80
OPEN(UNIT=IDISK,DEVICE='DSK',ACCESS='SEQOUT')	PRNC 85
IDEV=IDISK	PRNC 90
GO TO 60	PRNC 95
50 IF (ID.NE.'LPT') GO TO 20	PRNC 100
IDEV=3	PRNC 105
OPEN(UNIT=3,DEVICE='LPT',ACCESS='SEQOUT')	PRNC 110
60 DO 80 I=1,61	PRNC 115
IPLOT(I,1)=0	PRNC 120
DO 70 J=2,ICOL+2	PRNC 125
IPLOT(I,J)=KBLK	PRNC 130
70 CONTINUE	PRNC 135
80 CONTINUE	PRNC 140
DO 90 J=2,ICOL+2	PRNC 145
IPLOT(1,J)=KSTAR	PRNC 150
IPLOT(61,J)=KSTAR	PRNC 155
90 CONTINUE	PRNC 160
DO 110 I=2,60	PRNC 165
IPLOT(I,2)=KSTAR	PRNC 170
IPLOT(I,ICOL+2)=KSTAR	PRNC 175
DO 100 J=7,ICOL-3,5	PRNC 180
IPLOT(I,J)=KGRID	PRNC 185
100 CONTINUE	PRNC 190
110 CONTINUE	PRNC 195
IF (ITYPE.NE.1) GO TO 140	PRNC 200
120 WRITE(6,10600)	PRNC 205
WRITE(6,11600)	PRNC 210
READ(6,10700)YORG	PRNC 215
IF (YORG.LT.-1E8.OR.YORG.GT.1E8) GO TO 120	PRNC 220
WRITE(6,10800)	PRNC 225
WRITE(6,11300)	PRNC 230
READ(6,10700)YSTEP	PRNC 235
DO 130 I=1,61,5	PRNC 240
IPLOT(I,2)=KTIC	PRNC 245
130 CONTINUE	PRNC 250
CALL LINPLT(IPLOT,YORG,YSTEP,PARM,ICOL,INP,B)	PRNC 255
CALL PRNPLT(IPLOT,ITYPE,YORG,ICYCLE,YSTEP,TITLE,KYLAB,IDEV,ICOL)	PRNC 260
RETURN	PRNC 265

```

140 IF (ITYPE.EQ.2) GO TO 150
WRITE(6,11200)
RETURN
150 WRITE(6,10600)
WRITE(6,11700)
160 WRITE(6,10900)
READ(6,10700)YORG
IF (YORG.EQ.0.0) GO TO 160
IF (YORG.LT.1E-15.OR.YORG.GT.1E8) GO TO 150
170 WRITE(6,11000)
READ(6,11100)ICYLE
IF (ICYLE.LT.1.OR.ICYLE.GT.5) GO TO 170
!EXP=INT(ALOG10(YORG))
CYOR=YORG/10.**!EXP
CYOR=FLOAT(INT(CYOR))
ACYOR=CYOR*10.**!EXP
190 !TIER=60/ICYLE
DO 210 K=0,ICYLE-1
  ORG=CYOR
  KOUNT=0
  MAX=10.
  ORFAC=ALOG(CYOR)+ALOG(10.)
190  DO 200 AL=ORG,MAX
    ALGTC=ALOG(10.)+ALOG(AL)
    CALL NORM(ALGTC,ORFAC,!TIER,ALOG(10.),IROW)
    IROW=IROW+(!TIER*K)
    !PLOT(IROW,2)=KTIC
    !PLOT(IROW,1)=AL
    KOUNT=KOUNT+1
200  CONTINUE
  OPG=1
  MAX=CYOR
  ORFAC=ALOG(CYOR)
  IF (KOUNT.LT.10) GO TO 190
210  CONTINUE
  CALL LOGPLT(!PLOT,ACYOR,ICYLE,PAPM,INP,B,ICOL)
  CALL PRNPLT(!PLOT,ITYPE,ACYOR,ICYLE,YSTEP,TITLE,KYLAB,IDEV,ICOL)
  RETURN
10000 FORMAT(/' ENTER Y-AXIS LABEL (20 CHARACTERS MAXIMUM)')
10100 FORMAT(A5)
10200 FORMAT(/' ENTER NUMBER OF COLUMNS FOR PLOT (60 OR 120)')
10300 FORMAT(I3)
10400 FORMAT(/' ENTER OUTPUT DEVICE (TTY, DSK, OR LPT)')
10500 FORMAT(A3)
10600 FORMAT(/' ENTER Y-AXIS ORIGIN')
10700 FORMAT(G)
10800 FORMAT(/' ENTER Y-AXIS STEP SIZE (UNITS/5 LINES OF PLOT,')
10900 FORMAT(' (CANNOT BE ZERO)')
11000 FORMAT(/' ENTER NUMBER OF CYCLES FOR LOG PLOT (INTEGER,1-5)')
11100 FORMAT(I1)
11200 FORMAT(/' UNRECOGNIZED PLOT TYPE : ERROR IN MAIN')
11300 FORMAT(' 60 LINES OF PLOT')
11400 FORMAT(/' ENTER DISK FILE NUMBER (7-63)')
11500 FORMAT(I2)
11600 FORMAT(/' BETWEEN -1E8 AND 1E8')
11700 FORMAT(/' BETWEEN 1E-15 AND 1E8')
END

```

```

PRNC 270
PRNC 275
PRNC 280
PRNC 285
PRNC 290
PRNC 295
PRNC 300
PRNC 305
PRNC 310
PRNC 315
PRNC 320
PRNC 325
PRNC 330
PRNC 335
PRNC 340
PRNC 345
PRNC 350
PRNC 355
PRNC 360
PRNC 365
PRNC 370
PRNC 375
PRNC 380
PRNC 385
PRNC 390
PRNC 395
PRNC 400
PRNC 405
PRNC 410
PRNC 415
PRNC 420
PRNC 425
PRNC 430
PRNC 435
PRNC 440
PRNC 445
PRNC 450
PRNC 455
PRNC 460
PRNC 465
PRNC 470
PRNC 475
PRNC 480
PRNC 485
PRNC 490
PRNC 495
PRNC 500
PRNC 505
PRNC 510
PRNC 515
PRNC 520
PRNC 525
PRNC 530
PRNC 535
PRNC 540
PRNC 545
PRNC 550

```

```

SUBROUTINE PRNPLOT(IPLT,ITYPE,YORG,ICYCLE,YSTEP,TITLE,KYLAB,IDEV,PRNP 0
1 ICOL) PRNP 5
DIMENSION IPLOT(61,122),TITLE(5),KYLAB(4) PRNP 10
IF (ICOL.EQ.120) GO TO 10 PRNP 15
WRITE(IDEV,10000)(TITLE(I),I=1,4) PRNP 20
WRITE(IDEV,10100)(KYLAB(I),I=1,4) PRNP 25
GO TO 20 PRNP 30
10 WRITE(IDEV,10200)(TITLE(I),I=1,4) PRNP 35
WRITE(IDEV,10300)(KYLAB(I),I=1,4) PRNP 40
20 CONTINUE PRNP 45
IF (ITYPE.NE.1) GO TO 60 PRNP 50
IWLAB=5 PRNP 55
YLAB=YORG+(12*YSTEP) PRNP 60
DO 50 I=61,1,-1 PRNP 65
IF (IWLAB.NE.5) GO TO 30 PRNP 70
IWLAB=0 PRNP 75
WRITE(IDEV,10400)YLAB,(IPLOT(I,J),J=2,ICOL+2) PRNP 80
YLAB=YLAB-YSTEP PRNP 85
GO TO 40 PRNP 90
30 WRITE(IDEV,10500)(IPLOT(I,J),J=2,ICOL+2) PRNP 95
40 CONTINUE PRNP 100
IWLAB=IWLAB+1 PRNP 105
50 CONTINUE PRNP 110
GO TO 110 PRNP 115
60 CONTINUE PRNP 120
ITIER=60/ICYCLE PRNP 125
IWLAB=ITIER PRNP 130
YLAB=YORG+10**ICYCLE PRNP 135
DO 100 I=61,1,-1 PRNP 140
IF (IWLAB.NE.ITIER) GO TO 70 PRNP 145
WRITE(IDEV,10400)YLAB,(IPLOT(I,J),J=2,ICOL+2) PRNP 150
IWLAB=0 PRNP 155
YLAB=YLAB/10 PRNP 160
GO TO 90 PRNP 165
70 IF (IPLOT(I,1).EQ.0) GO TO 80 PRNP 170
WRITE(IDEV,10600)(IPLOT(I,J),J=1,ICOL+2) PRNP 175
GO TO 90 PRNP 180
80 WRITE(IDEV,10500)(IPLOT(I,J),J=2,ICOL+2) PRNP 185
90 CONTINUE PRNP 190
IWLAB=IWLAB+1 PRNP 195
100 CONTINUE PRNP 200
110 IF (ICOL.EQ.120) GO TO 120 PRNP 205
WRITE(IDEV,10700) PRNP 210
WRITE(IDEV,10800) PRNP 215
GO TO 130 PRNP 220
120 WRITE(IDEV,10900) PRNP 225
WRITE(IDEV,11000) PRNP 230
130 CLOSE(UNIT=IDEV) PRNP 235
RETURN PRNP 240
10000 FORMAT(29X,4A5) PRNP 245
10100 FORMAT(5X,4A5) PRNP 250
10200 FORMAT(59X,4A5) PRNP 255
10300 FORMAT(5X,4A5) PRNP 260
10400 FORMAT(1X,1PE10.3,121A1) PRNP 265
10500 FORMAT(11X,121A1) PRNP 270
10600 FORMAT(9X,12,121A1) PRNP 275
10700 FORMAT(10X,'.14',2X,'.62',2X,'2.3',2X,'6.7',3X,'16',3X,'31',3X, PRNP 280
1 '50',3X,'69',3X,'84',3X,'93',2X,'97.7',1X,'99.4',1X,'99.9') PRNP 285
10800 FORMAT(25X,'CUMULATIVE PROBABILITY (X)') PRNP 290

```

```

10900 FORMAT(10X,'.14',7X,'.62',7X,'2.3',7X,'6.7',7X,'16', 8X,'31',8X, PRNP 295
1 '50',8X,'69',8X,'84',8X,'93',7X,'97.7',6X,'99.4',5X,'99.9') PRNP 300
11000 FORMAT(57X,'CUMULATIVE PROBABILITY (%)') PRNP 305
END PRNP 310

```

```

SUBROUTINE UERTST (IER,NAME) UERT 0
C UERTST-----LIBRARY 1----- UERT 5
C UERT 10
C FUNCTION - ERROR MESSAGE GENERATION UERT 15
C USAGE - CALL UERTST(IER,NAME) UERT 20
C PARAMETERS IER - ERROR PARAMETER. TYPE + N WHERE UERT 25
C TYPE= 128 IMPLIES TERMINAL EPRDP UERT 30
C 64 IMPLIES WARNING WITH FIX UERT 35
C 32 IMPLIES WARNING UERT 40
C N = ERROR CODE RELEVANT TO CALLING ROUTINE UERT 45
C NAME - INPUT VECTOR CONTAINING THE NAME OF THE UERT 50
C CALLING ROUTINE AS A SIX CHARACTER LITERAL UERT 55
C STRING. UERT 60
C LANGUAGE - FORTRAN UERT 65
C----- UERT 70
C LATEST REVISION - JANUARY 18, 1974 UERT 75
C UERT 80
C UERT 85
C UERT 90
C DIMENSION ITP(5,4),IBIT(4) UERT 95
C INTEGER NAME(3) UERT 100
C INTEGER WARN,WARNF,TERM,PRINTR UERT 105
C EQUIVALENCE (IBIT(1),WARN),(IBIT(2),WARNF),(IBIT(3),TERM) UERT 110
C DATA ITP /'WARN','ING',,,'',,,'WARN','ING',, UERT 115
C 1 'WITH','FIX',,,'', 'TERM','INAL',,,'',,,'NON-', UERT 120
C 1 'DEFI','NED',,,'',,,'/' UERT 125
C IBIT(1)=32 UERT 130
C IBIT(2)=64 UERT 135
C IBIT(3)=128 UERT 140
C IBIT(4)=0 UERT 145
C DATA PRINTR / 6/ UERT 150
C IER2=IER UERT 155
C IF (IER2.GE.WARN) GO TO 10 UERT 160
C NON-DEFINED UERT 165
C IER1=4 UERT 170
C GO TO 40 UERT 175
C 10 IF (IER2.LT.TERM) GO TO 20 UERT 180
C TERMINAL UERT 185
C IER1=3 UERT 190
C GO TO 40 UERT 195
C 20 IF (IER2.LT.WARNF) GO TO 30 UERT 200
C WARNING(WITH FIX) UERT 205
C IER1=2 UERT 210
C GO TO 40 UERT 215
C WARNING UERT 220
C 30 IER1=1 UERT 225
C EXTRACT 'N' UERT 230
C 40 IER2=IER2-IBIT(IER1) UERT 235
C PRINT ERROR MESSAGE UERT 240

```

```

      WRITE(PRINTR,10000) (ITYP(I,IER1),I=1,5),NAME,IER2,IER
      RETURN
10000 FORMAT(' *** ! M S L(UERTST) *** ',5A4,4X,3A2,4X,I2, ' (IER = ',
1 I3,'')
      END

```

UERT 245

UERT 250

UERT 255

UERT 260

UERT 265

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

APPENDIX F

Input Data Set for Sample Run

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

46.0
47.7
37.3
58.6
45.4
29.3
60.9
62.2
62.2
70.4
51.1
60.6
47.9
58.8
29.7
49.6
56.6
57.3
29.8
46.0
42.1
37.0

THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

APPENDIX G

Modification for Systems Without DISSPLA Graphics Software

**THIS PAGE
WAS INTENTIONALLY
LEFT BLANK**

APPENDIX G. Modification for Systems Without DISSPLA Graphics Software

There several modifications that can be made to allow TERPED/P to execute on a computer system that does not have the DISSPLA graphics package. These modifications simply remove calls to the missing DISSPLA subroutines. They are presented in order of occurrence within the program, starting with the reference number of each statement (found in columns 72-80), then the type of modification (delete, add, or change statement), and if a change, how the new statement reads.

- 1) TERP 55 Delete
- 2) TERP 225 - TERP 245 Delete lines
- 3) TERP 270 - TERP 285 Delete lines
- 4) TERP 265 change 70 STOP
- 5) DSPL Ø - DSPL 385 Delete lines (remove entire subroutine)

If desired, the FORMAT statement at TERP 305 - TERP 310 can be changed to reflect the missing high-resolution plot generator.

These modifications in no way affect the operation of the rest of the code. If the 'H' character is entered to start the high-resolution plotter after all the modifications have been completed, it is treated as invalid input and the user is requested to re-enter a valid command.

**THIS PAGE
WAS INTENTIONALLY
LEFT BLANK**

ORNL-5813
Dist. Category
UC-11, UC-41

INTERNAL DISTRIBUTION

1. C. F. Baes, III	37. H. R. Meyer
2. J. J. Beauchamp	38. C. W. Miller
3. R. O. Chester	39. R. V. O'Neill
4. S. J. Cotter	40. D. C. Parzyck
5. C. Coutant	41. M. R. Patterson
6. W. G. Craddick	42. R. M. Reed
7. M. I. Dyer	43. D. E. Reichle
8. K. F. Eckerman	44. P. S. Rohwer
9. C. J. Emerson	45. R. D. Sharp
10-20. D. E. Fields	46. G. W. Westley
21-31. S. R. Glandon	47. Central Research Library
32. S. J. Gull	48. ORNL Y-12 Technical Library
33. F. O. Hoffman	49. Laboratory Records
34. S. V. Kaye	50. Laboratory Records - RC
35. G. G. Killough	51. ORNL Patent Office
36. R. S. Lowrie	

EXTERNAL DISTRIBUTION

52. Office of Assistant Manager, Energy Research and Development, Department of Energy, Oak Ridge Operations Office, Oak Ridge, Tennessee 37830
- 53-336. Given distribution as shown in TID-4500 under category UC-11, Environmental Control Technology and Earth Sciences; and UC-41, Health and Safety