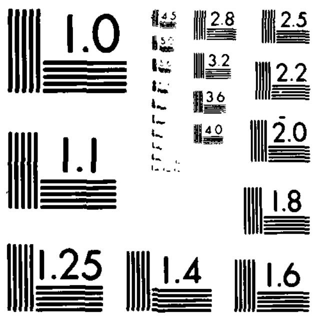
AD-A109 602 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 9/2
STATISTICAL ANALYSIS SOFTWARE FOR THE TRS-80 MICROCOMPUTER.(U)
SEP 81 R P ISBELL

UNCLASSIFIED NL

1 OF 2



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

STATISTICAL ANALYSIS SOFTWARE
FOR THE TRS-80 MICROCOMPUTER

by

Robert Paul Isbell

September 1981

Thesis Advisor: Charles F. Taylor, Jr.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A104602	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Stastical Analysis Software for the TRS-80 Microcomputer		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis September 1981
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert Paul Isbell		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE September 1981
		13. NUMBER OF PAGES 120
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

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

18. SUPPLEMENTARY NOTES

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

TRS-80
Microcomputer
Statistical Analysis
Probability Distributions
Hypothesis Tests
Confidence Interval
Multiple Linear Regression
Analysis of Variance
Data Plots

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

This paper documents the development of a statistical analysis package for the TRS-80 microcomputer. The package is comprised of six interactive programs which are generally divided into topical areas. The major emphasis is on exploratory data analysis and statistical inference, however, probability and inverse probability distributions are also included.

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

The programming language is TRS-80 Level II BASIC enhanced by the input/output commands available through the ESF-80 (Exatron Stringy Floppy) mass storage subsystem. With the modification of these few commands, the package is compatible with most floppy disk operating systems designed for the TRS-80 Model I or Model III microcomputers.

This statistical analysis capability implemented on a relatively inexpensive system provides a useful tool to the student or the trained analyst without ready access to a mainframe computer system.

Approved for public release, distribution unlimited

Statistical Analysis Software for the TRS-80 Microcomputer

by

Robert Paul Isbell
Major, United States Marine Corps
B. S., United States Naval Academy, 1967

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1981

Author Robert Paul Isbell

Approved by: Charles F. Taylor, Jr., Thesis Advisor

Donald R Barr, Second Reader

Chairman, Department of Operations Research

Dean of Information and Policy Sciences

ABSTRACT

This paper documents the development of a statistical analysis package for the TRS-80 microcomputer. The package is comprised of six interactive programs which are generally divided into topical areas. The major emphasis is on exploratory data analysis and statistical inference, however, probability and inverse probability distributions are also included.

The programming language is TRS-80 Level II BASIC enhanced by the input/output commands available through the ESF-80 (Exatron Stringy Floppy) mass storage subsystem. With the modification of these few commands, the package is compatible with most floppy disk operating systems designed for the TRS-80 Model I or Model III microcomputers.

This statistical analysis capability implemented on a relatively inexpensive system provides a useful tool to the student or the trained analyst without ready access to a mainframe computer system.

TABLE OF CONTENTS

I. INTRODUCTION

Over the past decade computers have played an ever increasing role in the day to day operation and administration of the military services. In the Marine Corps, for example, computer-based systems such as MMS (Manpower Management System), JUMPS (Joint Uniform Military Pay System), MIMMS (Marine Corps Integrated Maintenance Management System), and many others handle virtually all personnel administration and personnel pay matters, equipment maintenance and operational status reporting, supply requisitioning, fiscal record-keeping, etc. The computers which support these systems are large mainframe devices usually shared by two or more major commands. While the small unit commander certainly benefits from the overall effects of these large computer systems, he has little or no access to a computer system for projects tailored to his individual unit.

The small, relatively inexpensive microcomputer systems which abound today offer the small unit commander the possibilities of a customized computer based system for unit level information processing, record keeping and problem solving. The hardware is available and within the budget of these lower levels of command but will be of little use unless appropriate software is also available.

The purpose of this thesis is to provide a software package consisting of a number of statistical analysis procedures which can be effectively implemented on a microcomputer.

II. OVERVIEW

A. PACKAGE CONTENT

McNeil [Ref. 1] states that an effective data analyst requires as his basic tools a computer and a set of routine procedures. By his definition, the computer can range from the statistican with pencil and paper to a wide variety of electronic computational devices. Whatever form it takes, the function of the computer is to faithfully execute the collection of routine procedures selected by the analyst. By implication, the key to an effective statistical analysis package is the proper choice of procedures to be included.

The collection of "routine procedures" contained in this package are intended to provide the potential user with a basic set of widely utilized statistical analysis tools. These procedures have broad application and are not designed for any one specific scenario. The package was, however, designed with a specific user in mind, i.e. the military officer with a working knowledge of statistical analysis techniques and a desire to apply these techniques at the small unit level. The potential applications remain broad but typically include such items as forecasting next quarter's training support requirements, temporary additional duty budget, spare parts demand, etc., without complete knowledge of the number of personnel who will be assigned, what changes might be made in exercise schedules, equipment usage rate and many other pertinent facts. In other words, he must make plans, decisions, and forecasts based upon his best estimation of the values of random variables. A common sense approach to this problem is to collect data on previous observations of the random variable in question and use this data as a indicator of what can be expected in the future.

In order to make knowledgeable decisions based on the available data it is first necessary to gain insight into the nature and structure of the data. Procedures utilized for this purpose are generally classified as exploratory data analysis techniques. A number of the procedures included in the package fall under this general classification. These include subroutines which calculate sample statistics such as sample mean, sample variance, skewness, kurtosis and the seven number summary. Also included are data plots such as histogram, stem and leaf, normal probability plot and exponential probability plot. These procedures were selected because the manner in which they represent the data is straightforward and easily interpreted by the trained analyst.

After the analyst has studied the nature and structure of the data he is usually interested in knowing what inferences concerning various population parameters can be drawn from the data. Statistical analysis procedures which are concerned with this question fall under the heading or statistical inference. Statistical inference can be further subdivided. If, through exploratory data analysis of by other means, the analyst feels that it is reasonable to assume knowledge of the distributional family from which the data sample is drawn, then, according to DeGroot [Ref. 2], he is dealing in the area of parametric statistical inference. If, on the other hand, inferences are made about population parameters without assuming an underlying distributional form, the term nonparametric statistical inference is applicable. Procedures which rest on implicit population assumptions but which are tolerant of departures from these assumptions are called robust procedures.

Parametric statistical inference procedures contained in this package include tests of hypotheses and confidence intervals for population parameters under the normal, exponential and Bernoulli distributional assumptions. While these three distributions are not the only ones of potential interest, they are probably the most common distributions assumed.

Procedures which provide robust estimates for population parameters are also included in the package. The robust-t (Winsorized-t and trimmed-t) procedures are utilized for establishing a point estimate and confidence interval for the population mean. The Jackknife technique is used for a point estimate and interval estimate for the population variance. The one true nonparametric statistical inference procedure included in the package is a subroutine which computes a confidence interval for the population median.

As implemented in this package, the procedures thus far mentioned deal with univariate data, i.e. data which represents a single random variable. Also included is a multiple linear regression procedure which, using the method of ordinary least squares, tests for a statistically significant linear relationship between a dependent (response) variable and one or more independent (carrier) variables. The regression equation coefficients which estimate the linear relationship and a number of statistics which measure the "goodness" of the fit are also calculated. Linear regression is one of the most widely utilized statistical analysis procedures and is a powerful tool when properly used. A typical application for a military unit might be to study the relationship between the number of Not Operationally Ready (NOR) pieces of equipment reported for each month over some period of time (the response variable) and such carriers as the average number of technicians assigned for the month, number of days spent in the field during the month, average response time for supply demands, etc.

ANalysis Of VAriance (ANOVA) procedures, which are used to test for statistically significant differences between two or more population means, are also included. This program allows for one-way classification of populations, e.g. scores on the GMS test (General Military Subjects) made by individuals classified according to MOS (Military Ocupational Speciality); two-way classification, e.g. GMS scores made by individuals classified according to MOS and sex; or three-way classification, e.g. GMS scores made by individuals classified according to MOS, sex and highschool completion status. As in the case of multiple linear regression, the ANOVA procedure is a very powerful and widely utilized analysis tool.

Finally, the package contains several of the more common discrete and continuous probability distributions and inverse distributions and three common data transformations. The probability distributions and inverse probability distributions are of interest to the analyst in their own right and many also serve as subroutines for other statistical analysis procedures contained in the package. The data transformations (square root, cube root and natural logarithm) can be useful during the exploratory data analysis phase in helping to uncover structure in the data.

Except in the more trivial cases, e.g. sample mean, the procedures mentioned above and the algorithms utilized to implement the procedures will be discussed in more detail in later sections of this paper. A summary of the procedures contained in this package is given in Figure 1.

B. TRS-80 MICROCOMPUTER SYSTEM

As previously stated, the analyst can accomplish the calculations required to perform the statistical analysis procedures outlined above in a variety of ways. These range from pencil and paper to large mainframe computer systems. The computations required for very small data sets can easily be handled with pencil and paper and perhaps a handheld calculator while the practical considerations of time alone dictates that very large data sets require a mainframe computer system for effective analysis. It is analysis of the moderate size data set (perhaps twenty to several hundred data points) which would become quite tedious to accomplish by hand but which might not merit the expense of a large computer system, assuming one were available, that the microcomputer handles nicely.

A number of microcomputers are currently available and new makes and models are appearing rapidly. This paper makes no attempt to draw qualitative comparisons between the competing brands. No such study was undertaken. The TRS-80 microcomputer was selected for this project simply because it appeared to offer capabilities comparable to the other popular makes at a slightly lower price. In addition, at least at the time the computer was purchased, the TRS-80

BLOCK DIAGRAM OF PACKAGE

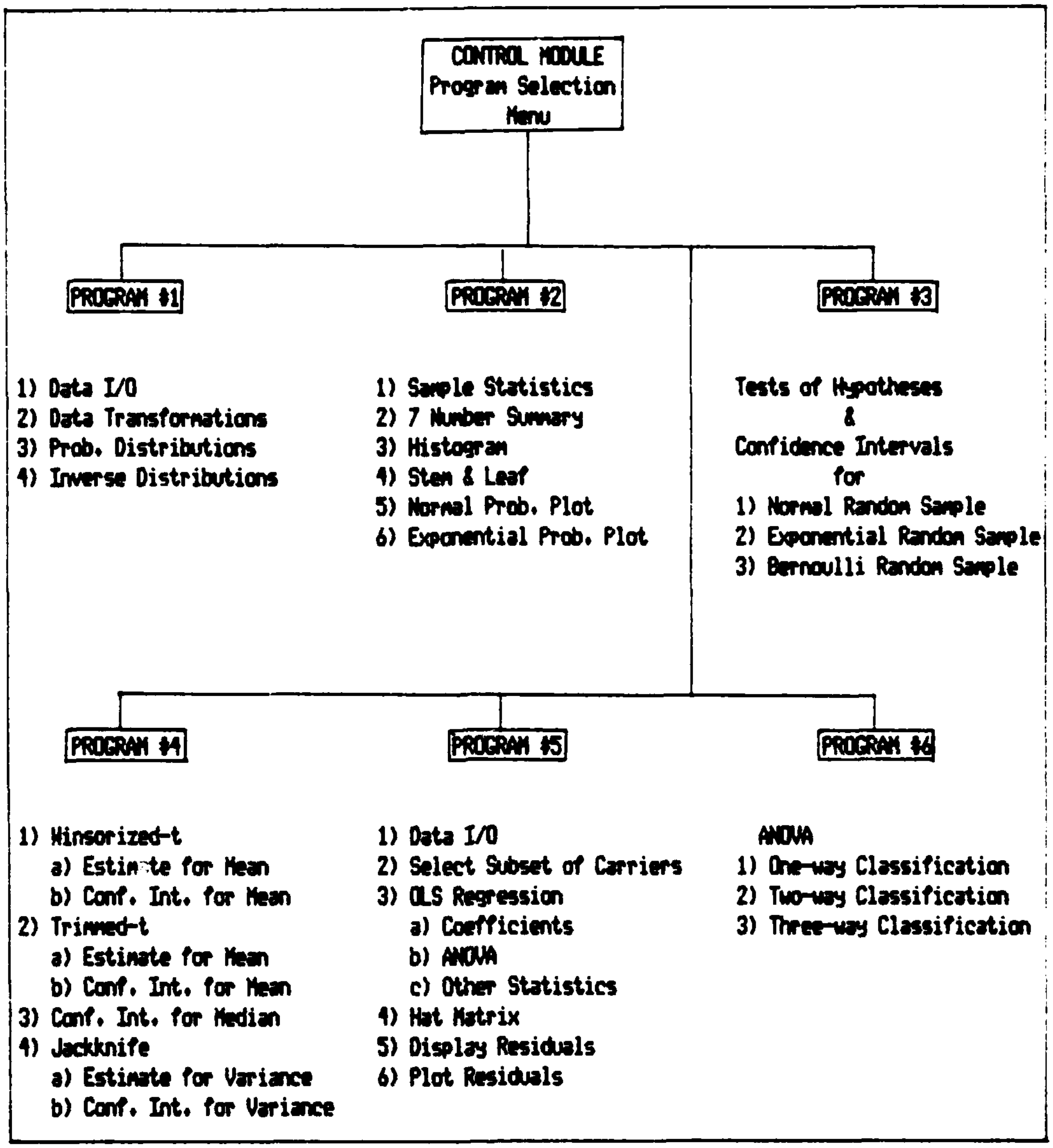


FIGURE 1

appeared to have the widest market acceptance. A more detailed description of the system follows.

This package was developed on a TRS-80, Model I, Level II microcomputer. The basic system consists of a 16K RAM (Random Access Memory) CPU/Keyboard (Central Processing Unit), a 12 inch diagonal video display, and a cassette recorder for loading and recording programs and data. This basic system was enhanced with a Microtek 32K RAM memory expansion unit and a ESF-80 (Exatron Stringy Floppy) mass storage subsystem which was used in lieu of the cassette recorder. The entire system cost less than $1,400.

This system proved to be reliable and provided the capabilities necessary for the successful completion of this project. A major enhancement, however, would have been at least one floppy disk system for loading programs and for input/output of data files. This addition would have significantly increased the speed of program loading and data file handling and provided the potential of alternate programming languages.

C. PROGRAMMING LANGUAGE

The programming language for this package is TRS-80 Level II BASIC developed by Microsoft Inc. A Level II BASIC interpreter is resident in the ROM (Read Only Memory) contained in the TRS-80 CPU. This "built in" language was used as an expediency; the comparatively slow execution speed of BASIC is not a significant factor for problems of the size this package is designed to handle.

The native Level II BASIC was enhanced with several system commands made available through the ROM contained in the ESF-80 mass storage subsystem. These commands accomplish several fundamental operations. The @SAVE and @LOAD commands are used to record and load program code. The @OPEN, @CLOSE, @PRINT and @INPUT commands are for data file handling. Finally, inclusion of the @LOAD command in program code enables the chaining of programs (the ability to load a program while preserving the variables created by the previously loaded programs).

D. PACKAGE PORTABILITY

Without at least some minor modifications this package is portable only to an identical system. Virtually all DOS's (Disk Operating Systems) designed for the TRS-80 Model I and Model III microcomputers, however, contain as a subset system commands which perform the same functions as those provided by the ESF-80 subsystem. Modification of these commands, which are contained in perhaps six lines of code in each of the six programs in the package, would make it portable to any TRS-80 Model I or Model III 48K RAM system equipped with at least one disk drive system.

The most significant differences between Level II BASIC and the versions of BASIC implemented on other microcomputers are keywords relating to display output. Because of the large amount of code directly relating to display output, it would require a sizeable effort to modify this package for another microcomputer. The program logic, however, could serve as a model for programs written in other versions of BASIC or even FORTRAN. A partial translation of this package into FORTRAN-IV was accomplished for implementation on a PDP-11-60 minicomputer system.

E. LIMITATIONS

Three interrelated constraints of this package are accuracy, speed and capacity. TRS-80 Level II BASIC includes the capability of defining double precision variables, thus significantly increasing the accuracy of computations. Double precision variables, however, require increased memory for storage and computation, with a resultant decrease in the capacity of the computer to handle large data arrays, and an increase in computation time. On the other hand, when variables are defined as single precision some algorithms which use approximations requiring increased precision must be rejected in favor of others which are often slower. The goal then was to strike a proper balance so that a good degree of accuracy could be attained and, at the some time, allow for reasonable speed of execution and capacity.

The majority of the variables in this package are single precision variables, but integer variables were used where possible and a few procedures require the use of string, i.e. character or literal, variables. No double precision variables are used. Over the range of problems tested, the accuracy was found to be good (four or more decimal place accuracy) in most instances and fair (two or three decimal place accuracy) in a few cases.

The capacity of the package is limited to a vector of 200 observations for those procedures which apply to univariate data, i.e. the first four programs. The multiple linear regression program is limited to a data array of 10X100 (response variable, 9 carrier variables and 100 observations). The ANOVA program is limited to 10X20, 10X10X10, and 10X10X10 arrays respectively for the one, two and three-way classification cases. The size of the data arrays handled by each program could be significantly increased if each program were loaded and executed as a single program. In order to chain the programs together and therefore pass variables back and forth between programs, the array sizes are limited to those cited above.

The package was designed for analysis of relatively small data arrays. Even so, due to the inherent nature of an interpreted language such as TRS-80 Level II BASIC, subroutines which require many iterative calculations, e.g. several of the inverse probability distributions and some of the robust procedures, can require up to several minutes of execution time. Each algorithm was designed to execute as fast as possible, but whenever a choice had to be made between speed of execution and accuracy, the more accurate algorithm was selected.

It is perhaps indicative of the current high degree of computational power available that an algorithm is considered slow when it requires several minutes of execution time. A decade ago analysis of a moderate size sample which did not merit very expensive computer time could have consumed hours of tedious hand calculations.

F. USER FRIENDLY FEATURES

This package was designed to be used by the individual with a general understanding of statistical analysis procedures and terminology. At the same time, it was recognized that there does not exist a completly standardized set of conventions for symbols and terms. While no attempt was made to define all procedures, parameters or terms, a brief explanation was provided where, in the subjective opinion of the author, it was felt confusion might exist. For example terms such as mean, variance, confidence level, sum of squares, F-statistic, etc. were assumed to be well understood by any potential user. When the user is initially introduced to the term "exponential parameter lambda", however, he is advised that the term refers to the reciprocal of the exponential mean.

Even the most knowledgeable user can inadvertently enter a value or character which would cause the program to terminate due to error or at least return an incorrect answer. Whenever the user is prompted for an input the value or character entered is first tested to see that it has some logical meaning within the context of the operation or calculation to be executed. If it is recognized that it does not then the user is so advised and is allowed to reenter the value or character. If, for example, a data transformation or a data plot procedure is selected after summary statistics have been entered, the user is advised that the selected subroutine is not applicable since data have not been entered and he is prompted for another selection.

III. DISCUSSION OF PROGRAMS/ALGORITHMS

A. STRUCTURE AND ORGANIZATION

The package is divided into six chained programs. The control module is integrated into program 1 and must always be loaded first. The control module serves to define, dimension, and initialize variables for the first four programs and positions the variable pointer so that subsequent programs can be loaded without loss of variables. The control module also contains the program instructions and program selection menu.

After initial execution of the control module any of the six programs may be selected by the user. A video prompt tells the user which program tape to load into the ESF-80; the program is then automatically loaded and begins execution (auto run). The user is then free to move from one program to the next with the following limitation: when either program 5 or program 6 is loaded all variables are cleared and redimensioned in order to accomodate the multidimensional arrays required by these programs. If the user desires to access one of the first four programs after using program 5 or program 6, it is necessary to first load program 1 and thus the control module in order to reinitialize variables.

B. PROGRAM 1

Program 1 contains data input/output procedures, data transformations and a number of common probability distributions and inverse probability distributions. A detailed block diagram of program 1 is given in figure 2.

1. Data Input/Output

Data Input/Output (I/O) for the first four programs is accomplished in program 1. Programs 5 and 6 contain their own data I/O subroutines. A one dimensional array of up to 200 data points may be manually entered via the keyboard, or a previously saved (recorded) data array may be loaded via the ESF-80. As a third alternative, the user may select one of four random number generators (normal, exponential, uniform and Bernoulli) to create data. Finally, the user is allowed to enter summary statistics only. This last capability is duplicated in program 3 to facilitate testing of various hypotheses.

Any time data or summary statistics are entered, all previously entered data is cleared. The user is therefore given the option to save data on tape in the event that further analysis is desired.

All data are stored in two single dimensional arrays, X(i) and Y(i). Both are defined as single precision real arrays which are stored with seven digits of precision but are printed with a maximum of six digits of precision (if the seventh digit is 5 or greater then it is

BLOCK DIAGRAM OF PROGRAM #1

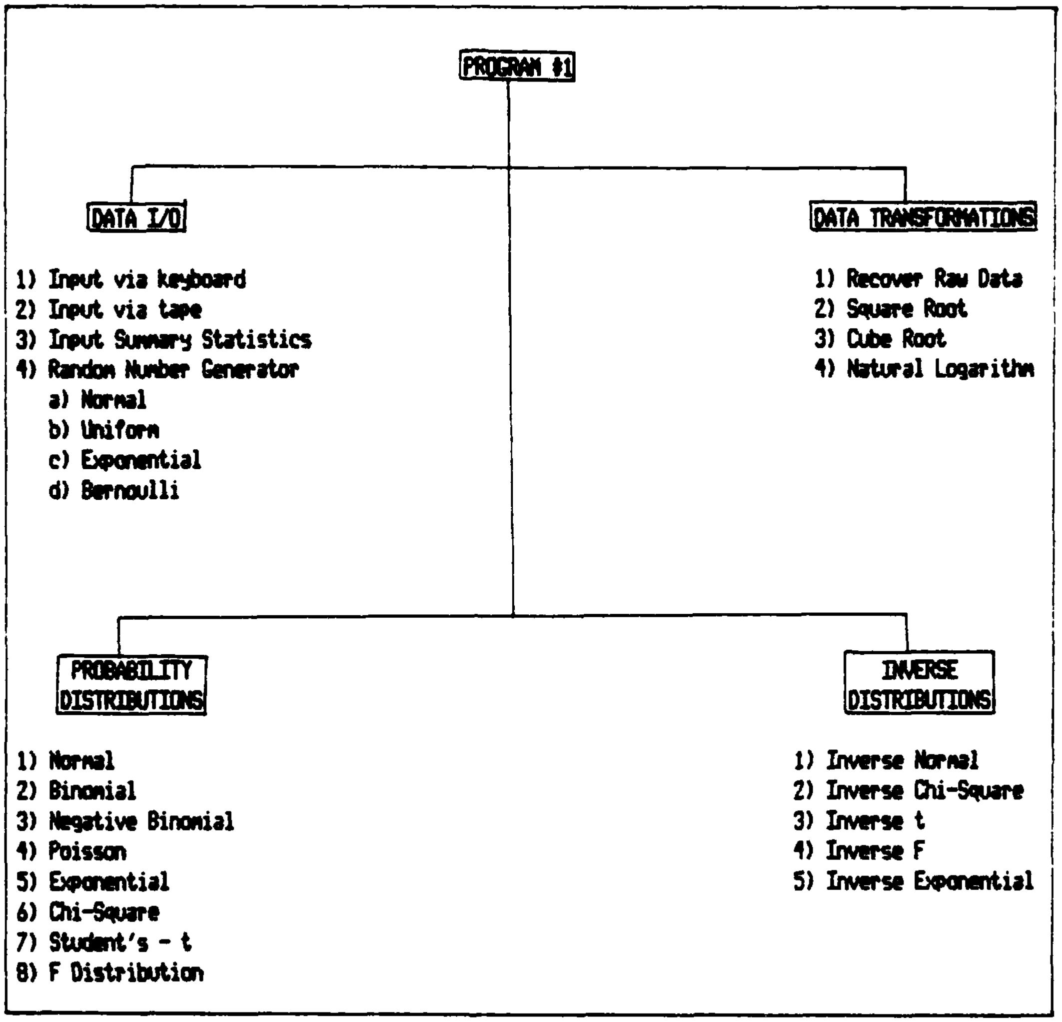


FIGURE 2

rounded up). All seven digits of precision are used for computations. The X array is used to save the raw data while the Y array may contain transformed data. All subroutines in the first four programs operate on the Y array.

After the data has been entered either manually via the keyboard, loaded via a prerecorded data tape or generated by one of the four random number generators, two operations on the arrays are automatically executed. First, unless the data are already ordered from low value to high value, the Shell Sort subroutine is called to order the data. After the data are sorted the following sample statistics are calculated and saved for later use: sample mean, sample variance (unbiased estimate of population variance), maximum likelihood estimate of variance, skewness and kurtosis.

2. Data Transformations

McNeil [Ref. 3] claims that data transformations are the most powerful tools avialable to the data analyst. Such adamant statements inevitably invite arguments but it is at least true that, when properly applied, data transformations can be very useful to the analyst. They can reveal structure in the data that is otherwise obscured and can reexpress the data so that assumptions such as normality are more nearly satisfied. The following data transformations are certainly not exhaustive but are three common transformations which have proved useful in data analysis work.

The square root, cube root and natural logarithm data transformations are optional after the raw data has been entered and sorted. The original data may be restored to the Y array from the X array by returning to the data transformations subroutine and selecting the 'No Transformation' option. After each data transformation sample statistics are recalculated.

3. Probability Distributions

The probability distributions menu contains five univariate continuous probability distributions, i.e. the normal, exponential, chi-square, Student's-t and the F distributions and three discrete probability distributions, the binomial, negative binomial and Poisson distributions. The discrete distributions calculate both point probability and cumulative probability while the continuous distributions calculate the left and right tail cumulative probabilities. The specific algorithms utilized are contained in the program listings at the end of this paper.

Kennedy and Gentle [Ref. 4] point out that 'rough' approximations which provide two or three figure accuracy over certain ranges of parameter values would be adequate if one were only interested in probabilities. There is a need for greater accuracy in statistical analysis packages, however, since these values are used in evaluating other expressions for which accurate results are required. A number of the probability distribution algorithms contained in program 1 are used as subroutines by other probability distributions, inverse probability distributions and, in later programs, used for evaluations such as confidence intervals and tests of hypotheses. Some of these applications will be discussed in following paragraphs.

In order to achieve the highest degree of accuracy possible, several of the probability distributions combine two or more algorithms. It is common that the faster algorithms work well over certain parameter ranges but lose accuracy quickly outside these ranges. It thus became necessary to utilize slower, often iterative algorithms in order to maintain an acceptable level of accuracy. Appendix A shows comparisons of probabilities calculated by the algorithms contained in this package and tabled values given in Handbook of Tables for Probability and Statistics [Ref. 5].

a. Normal Distribution

The central limit theorem simply stated says that the sample mean of a random sample drawn from any distribution with finite variance is, in the limit, normally distributed about the true population mean. This theorem makes the normal distribution one of the most important distributions in statistical analysis. A number of the statistical inference procedures contained in this package have the normal distribution as their foundation and several other probability distribution subroutines utilize relationships with the normal distribution.

The algorithm for the normal distribution is based on a polynomial expansion of the right tail cumulative probability given in Abramowitz and Stegun [Ref.6]. This algorithm gives results which are exact when compared to tabled values (see Appendix A). The nominal computation time is one second.

b. Exponential Distribution

The exponential distribution is commonly used to represent the distribution of time between the occurrences of certain events. One of the most frequent applications is in relaibilty theory.

The exponential cumulative probability is particularly easy and fast to calculate. The closed-form expression for the right tail cumulative probability is evaluated and the resulting probabilities are exact to single precision accuracy.

c. Chi-square Distribution

The importance of the chi-square distribution in statistical analysis is based on the theorem that the sum of squares of k independent and identically distributed standard normal random variables has a chi-square distribution with k degrees of freedom. Extensions of this theorem serve to test hypotheses concerning and establish confidence intervals for the population variance under the assumption of normality.

Two different algorithms were utilized to approximate chi-square cumulative probabilities. A relationship between the chi-square distribution and the normal distribution which is given in Abramowitz and Stegun [Ref. 7] was used for the large degrees of freedom case (degrees of freedom greater than thirty). This algorithm is fast (approximately one second) and is accurate to a minimum of three decimal places. For smaller degrees of freedom, however, the accuracy of this approximation falls off rapidly. Poole and Brochers [Ref. 8] give a BASIC program based on a series expansion of the right tail cumulative probability. This algorithm is slower (six seconds are required for thirty degrees of freedom) but maintains at least three decimal place accuracy down to one degree of freedom.

d. Student's - t Distribution

The t distribution, like the chi-square distribution, has wide application in statistical analysis work. By definition, if a standard normal random variable is divided by the square root of a chi-square random variable over its degrees of freedom then the resulting random variable has a t distribution with the same degrees of freedom. It can be shown that if a random sample of size n is drawn from a normal distribution then the square root of the sample size times the difference between the sample mean and true population mean all divided by the square root of the sample variance has a t distribution with n-1 degrees of freedom. An important point is that this t statistic does not depend on the true population variance.

Two algorithms were utilized to obtain t-distribution cumulative probabilities. Abramowitz and Stegun [Ref. 9] give a distributional relationship with the normal distribution which produces good results (three to four decimal place accuracy in one second) when degrees of freedom exceed thirty and a series expansion of the cumulative distribution function for smaller degrees of freedom. The latter algorithm returns four or more decimal place accuracy in one to three seconds.

e. F Distribution

The F distribution is still another sampling distribution of great importance in statistical analysis work. A random variable which is the ratio of two independent chi-square random variables each divided by its degrees of freedom has an F distribution with the respective numerator and denominator degrees of freedom. This statistic is often referred to as the F ratio and serves, for example, to test hypotheses concerning differences between variances of independent normal populations.

Kennedy and Gentle [Ref.10] point out that existing algorithms for F distribution probabilities do not give a high degree of accuracy. This reference does, however, give a FORTRAN program based on a series expansion of a basic integral form of the left tail cumulative probability which is among the best available algorithms. This package utilizes a direct translation of the FORTRAN program given in the above reference. The algorithm generally produces three decimal place accuracy and occasionally four decimal place accuracy. Computation time is a function of degrees of freedom and ranges from a nominal one second for small degrees of freedom to appoximately seven seconds for 120 numerator and denominator degrees of freedom.

f. Discrete Distributions

The binomial, negative binomial and Poisson distribution algorithms perform direct calculations of their respective probability functions to produce point probabilities. Cumulative probabilities are calculated by summation of point probabilities with one exception. The binomial cumulative probability utilizes the Poisson approximation if the number of trials is greater than 45 and the probability of occurrence on a single trial is either less than or equal to .1 or greater than or equal to .9. The normal approximation is used if the number of trials is greater than 45 and the probability of occurrence on a single trial times the number of trials is greater than 5.

All point probabilities are accurate to four or more decimal places. The normal and Poisson approximations to the binomial cumulative probability are accurate to three or more decimal places; the other cumulative probability procedures produce a minimum of four decimal place accuracy.

4. Inverse Probability Distributions

Exponential percentiles are calculated with the same ease as exponential probabilities and good closed form approximations for normal and chi-square percentiles were found. Abramowitz and Stegun [Ref. 11] was the source of the normal percentiles approximation while the algorithm for chi-square percentiles is a direct translation of FORTRAN program listed in "Collected Algorithms from ACM" [Ref. 12]. These algorithms give accurate results when compared to tabled values (see Appendix 2) and all execute in one to three seconds.

It was more difficult to obtain accurate results for t-distribution and F-distribution percentiles. Abramowitz and Stegun [Ref. 13] give closed form approximations for both distributions which perform well for larger degrees of freedom (ten or greater for the t-distribution and twenty/twenty or greater for the F-distribution). These approximations, however, lose accuracy rapidly for smaller degrees of freedom. In these cases it was necessary to resort to an algorithm which starts with a rough approximation of the percentile and, calling the respective probability distribution subroutine, iteratively refines the approximation until the returned probability converges to the specified value. This algorithm produces accurate percentiles but computation time can extend up to several minutes.

C. PROGRAM 2

Program 2 contains the six subroutines which are classified as exploratory data analysis procedures. Their primary purpose is to provide information about the data (raw or transformed) which reveals patterns, trends, spread, symmetry, how closely the data follows some classical distribution, etc. The information about the data is presented in two basic formats. The first two subroutines in this program present a numerical summary of the data while the last four subroutines give graphical representations (plots) of the data.

1. Numerical Summaries

The first numerical summary of the data displays several classical sample statistics, each of which is a measure of some facet of the data's structure. A measure of the central location and/or typical value of the data is given by the sample mean. Two measures of the spread of the data are given. The first is the sample variance, i.e. the unbiased estimate of population variance under the assumption of normality, and the second is the maximum likelihood estimate of population variance under the same assumption. The sample statistic called skewness provides a measure of the symmetry of the data. A value of zero indicates a symmetric distribution while positive and negative values indicate right and left skewed distributions respectively. Finally, kurtosis provides a measure of how much of the data is distributed into the tail of the distribution. For reference, a normal distribution has kurtosis of zero while the exponential distribution, which has a longer tail than the normal distribution, has kurtosis of six.

The second numerical summary of the data is the seven number summary. The "seven" refers to the basic elements of the display. These consist of the median, upper and lower quartiles (75 and 25 percentiles), the upper and lower eighths (87.5 and 12.5 percentiles) and the extremes (maximum and minimum).

Like the mean, the median gives a measure of the central or typical value of the data. The three remaining pairs of numbers give a measure of the spread of the data. Two other numbers associated with each of these pairs give a measure of skewness of the data. The first is simply the midspread or average of each of the pairs which can be compared with the median and with each other. The second measure of skewness makes this comparison a little more automatic. If M represents the median, Q1 the lower quartile, Qu the upper quartile then (((Qu-Ql)/2)-M)/(Qu-Ql) will be zero if the quartiles are symmetric about the median while a positive or negative number will suggest the corresponding direction of skewness. A similar number is calculated for eighths and for extremes.

2. Data Plots

The histogram is a graphical representation of the data in which each point is assigned to a class interval depending upon its value. The number of data points in a class interval determine the height of the graphical bar representing that cell. The resulting display is then a picture of the distribution which generally approximates the shape of the underlying probability distribution function.

An important parameter in determining the usefulness of the histogram is the width of the class interval. The extreme cases illustrate this point. If the class interval is so wide that all data points fall into a single cell then about the only information provided is the number of data points in the sample. On the other hand, if the class interval in so narrow that every data point belongs to a separate cell then the picture is very difficult to interpret. The histogram subroutine contained in this program can have up to fifteen cells. The initial class interval is automatically determined so that there will be a range of at least five cells from the minimum to the maximum data point. Some of these cells may be empty. After the initial histogram is drawn the user is given the option to increase the class interval or, subject to the fifteen cell limitation, decrease the class interval.

A graphical representation of the data similar to but containing more information than the histogram is the stem and leaf plot. The significant difference between the two plots is that the incremental elements in each cell of the stem and leaf, i.e. the leaves, are numbers instead of an incremental portion of a solid bar. The most significant digit of each data point is represented by the class interval labels running along the stem (vertical axis) of the plot and the leaves are the next most significant digit of each data point.

McNeil [Ref. 14] suggests several advantages the stem and leaf enjoys over the histogram. As well as providing the same information about the shape of the distribution it reveals intracell distribution. It is also possible to obtain reasonable estimates of percentiles simply by counting leaves.

As with the histogram, the initial class interval is automatically determined by the computer. After the first plot the user is again allowed to redraw the plot with a different class interval, this time subject to a limitation of sixteen cells.

A fairly simple procedure for testing the conformity of the data to some specified theoretical probability distribution is to plot the ordered sample against the corresponding theoretical percentiles. A one to one correspondence would, of course, plot as a straight line. Systematic departure from a straight line plot indicates that the sample does not conform well to the theoretical distribution.

This procedure was implemented for the normal and exponential distributions and these two plots are the final graphical representations of the data contained in this program. These plots perform well for fairly significant departures from the theoretical distributions. Due to the relatively low resolution graphics capability of the TRS-80 video display, however, slight to moderate departures from the theoretical distributions are difficult to detect.

D. PROGRAM 3

The statistical analysis procedures contained in program 3 are classified as parametric statistical inference procedures. They include classical tests of hypotheses and calculation of confidence intervals for normal, exponential and Bernoulli population parameters. The specifics of these classical procedures are given in almost all first course statistics textbooks and will not be reiterated here. The discussion will be limited to the basic format of the procedures. Major portions of each of the algorithms contained in this program have already been discussed, i.e.

the probability distributions and inverse distributions which were developed for program 1. A detailed block diagram of the hypotheses tested and confidence intervals calculated by this program is given in Figure 3.

1. Tests of Hypotheses

The basic format for the various test of hypotheses is as follows. Once the user has selected the assumed distribution and the specific null hypothesis to be tested then the appropriate test statistic (usually a combination of sample statistics already calculated in program 1) is calculated and displayed. Using this test statistic, the appropriate probability distribution subroutine is called and the probability at which this statistic will become statistically significant (P-value) is returned and displayed . The user is then prompted to enter the probability of type I error (alpha value) he desires to establish for the test. If alpha is less than the P-value then the null hypothesis is accepted, otherwise it is rejected.

A more classical approach to hypothesis testing is to calculate a value for a statistically significant statistic given the alpha value and then compare the actual test statistic calculated from the sample to this value to determine acceptance or rejection. This approach was rejected in favor of the one given above for two reasons. First the accuracy and speed of the probability distribution subroutines are generally better than those for the

BLOCK DIAGRAM OF PROGRAM #3

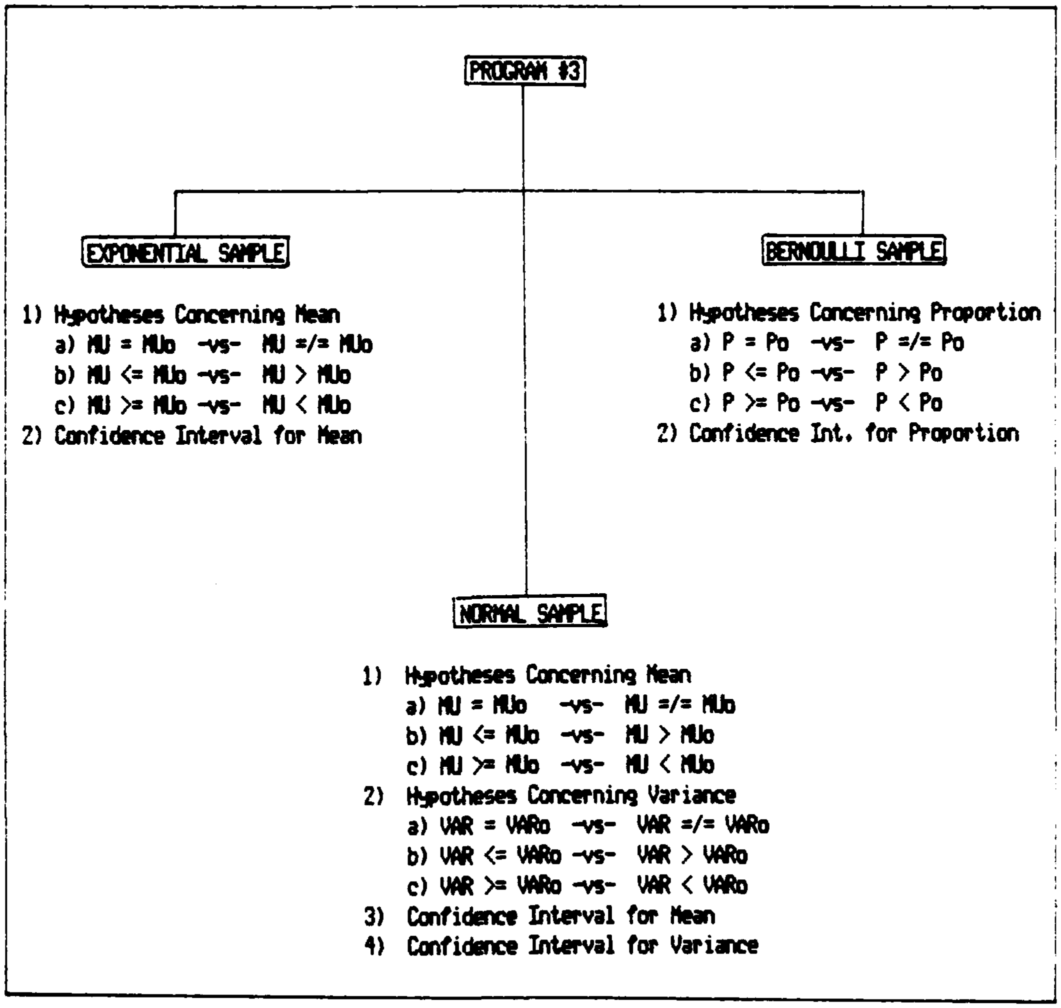


FIGURE 3

inverse distribution subroutines. Calculation of the P-value uses the probability distribution subroutines while calculation of a statistically significant value for the statistic uses the inverse probalility subroutines. Second, and perhaps most importantly, the displayed P-value provides more information than a simple accept/reject criteria.

2. Confidence Intervals

The format for the confidence interval subroutines is similar to that utilized for tests of hypotheses. The user selects the assumed population and the desired confidence level. After this information has been entered the appropriate inverse probability distribution is called and returns the value at which a corresponding sample statistic would become significant. A confidence bound for the parameter of interest is calculated from this value and displayed. This algorithm is first utilized to calculate two sided confidence limits and then a upper and lower one sided confidence limit.

E. PROGRAM 4

This program contains two robust and one nonparametric statistical inference procedures.

1. Robust-t

When it is suspected that the random sample contains outliers, then it is often desirable to obtain an estimate for the population mean that places less weight on the extreme points. Two techniques for obtaining such an estimate are the Winsorized-t and trimmed-t procedures. Winsorization is accomplished by symmetrically setting one or more of the extreme values of the ordered sample equal to its nearest neighbor. Trimming, on the other hand, omits the symmetrically placed extreme values. It should be noted that in both cases there is an implicit assumption that the true distributional form is symmetric. The depth to which the extreme values are Winsorized/trimmed is referred to as the level of Winsorization/trimming. After the data are Winsorized/trimmed the point estimate of the population mean is calculated in the usual manner. Dixon and Massey [Ref. 15] point out that there is little loss of efficiency if the data are Winsorized (at mild levels) when the extreme points are actually valid and that much improved estimates result when the extremes are not valid observations. Trimmed estimates are somewhat less efficient.

It has also been shown that the t-ratio which is calculated using the sample variance of the Winsorized data, multiplied by an appropriate constant (see Dixon and Massey [Ref. 16] for details) closely follows the t-distribution with h-1 degrees of freedom where h is the number of obervations not Winsorized/trimmed. This statistic is used to calculate confidence bounds for the true population mean.

This program allows Winsorizing/trimming of the data up to the tenth level. Computation time is a function of the selected level but is not excessive in any case.

2. Jackknife

If the distribution from which a random sample is drawn is normal then a confidence interval for the true population variance is properly based on the chi-square sampling distribution. Such confidence intervals are, however, very sensitive to the underlying distribution and do not perform well when the assumption of normality is violated. A robust method for handling departures from normality is the jackknife technique. The jackknife technique is discussed in some detail in Mosteller and Tukey [Ref. 17].

This procedure, more than any other in this package, shows up the speed disadvantage of an interpreted language like TRS-80 Level II BASIC. The crux of this procedure is to divide the data into g groups of size h and then iteratively calculate the sample variance g times, each time leaving out one group of h. The usual method is to let h=1 and therefore g=n where n is the size of the sample. This eliminates any arbitrariness in the formation of the groups but means that a sample of one-hundred data points requires one-hundred calculations of the sample variance. This specific example required 25 minutes of calculation time. As a partial solution to this problem, the user is allowed to set h to any integer between one and five so long as the data size is evenly divisible by h. When h is greater than one the ordered data must be randomized to avoid systematically leaving out the h smallest values first, then the next h smallest values, etc. Randomizing the data does introduce some of the arbitrariness mentioned earlier and results in slightly different answers each time the procedure is repeated. A number of sample problems, however, displayed reasonable stability from one calculation to the next. Application of the jackknife technique to a sample of one hundred observations divided into twenty groups of size 5 required approximately five minutes of calculation time.

3. Confidence Interval for the Median

By the definition of the median, the probability that any random observation drawn from any continuous population is less than or equal to the median equals one-half. Based on this fact it can be shown that the probability that the median lies between the ith and the jth order statistic of a random sample has a binomial distribution with parameters p=.5 and n equal to the sample size. Since no assumptions are made about the population from which the random sample is drawn other than that it is continuous, it is possible to obtain a truly nonparametric confidence interval for the population median.

Confidence intervals for the median calculated by this program are in exact agreement with tabled values contained in Dixon and Massey [Ref.18]. Calculation time varies from a few seconds to several minutes depending on the sample size.

F. PROGRAM 5

Program 5 contains a multiple linear regression procedure which, by the method of Ordinary Least Squares (OLS), estimates the coefficients of the basic linear model, Y = X B + e, where Y is the vector of observations of the dependent variable, X is the matrix of independent variables, B is the vector of coefficients estimated and e is the error vector. The program also calculates and displays a number of the standard statistics each of which measures some facet of the quality of the OLS solution.

The computational method used by this program is straightforward matrix algebra applied to the basic linear model. The greatest difficulty with this method is in obtaining accurate results when the X'X matrix is inverted. According to Draper and Smith [Ref. 19] one of the prime causes of roundoff error during this operation is the presence of widely differing orders of magnitude in the numbers involved. In order to minimize this problem the data are normalized or centered as described in the aforementioned reference. Two additional advantages result from normalizing the data. If Z represents the normalized X matrix then Z'Z is the correlation matrix and can be displayed to provide the analyst with additional information about the data. The coefficients which are directly estimated using the normalized data are referred to as the beta coefficients. The beta coefficients have the advantage of being independent of the units of measurement of the original data. This program displays both the standard regression equation coefficients and the beta coefficients.

Other standard statistics calculated and displayed are the ANOVA table statistics, R-square, adjusted R-square, standard error of estimate, partial-F for the coefficients and standard error of the coefficients. Each of these statistics with the exception of the adjusted R-square statistic is discussed in detail in Draper and Smith [Ref. 20]. The adjusted R-square statistic is discussed in Intriligator [Ref. 21].

A perhaps not so standard feature of this program is a display of the diagonal elements of the hat matrix. The hat matrix is defined as $X(X'X)^{-1}X'$, where these symbols have the same meaning as given earlier. The diagonal elements of this matrix give a measure of the influence or leverage of the ith observation of the response variable on the corresponding fitted value. Large values are indicative of high leverage observations and may indicate that these observations are outliers. Hoaglin and Welsch [Ref. 22] give as a rule of thumb for 'large' values, any value in excess of twice the rank of the hat matrix divided by the number of observations. All diagonal elements of the hat matrix are displayed and large values are highlighted.

This program allows the user to regress the dependent variable on all or any subset of the independent variables. It is therefore possible for the user to use various algorithms for selecting the 'best subset' of carriers.

The final feature of this program is a plot of the estimates of the response variable against the residuals. According to McNeil [Ref. 23] and other recognized authorities in the field of data analysis, such a display of the residuals is one of the best tools in existence for obtaining a measure of the overall fit of the data to the model.

The regression procedure contained in this program was tested against the Portland Cement example contained in Draper and Smith [Ref. 24]. The data are thirteen observations of a response variable and four carrier variables. This regression was completed in 48 seconds. The estimated regression equation coefficients and all statistics agreed with the reference to a minimum of three decimal places and in most cases four or more decimal places.

G. PROGRAM 6

Program 6 is the last program in this package and contains ANalysis Of VAriance (ANOVA) procedures. The algorithms for the ANOVA procedures were taken from Duncan [Ref. 25] who makes the claim that ANOVA is one of the most powerful tools in statistical analysis. He further states that the basic procedure of ANOVA consists of classifying and cross-classifying some statistical result of interest to determine if a specified classification is statistically important in affecting the result. A potential application of ANOVA was given in an earlier section of this paper.

It was attempted to include as much flexibility in the ANOVA procedures contained in this program as feasible. One, two and three-way classification of the statistical result of interest is allowed. The one-way classification procedure allows up to ten levels of the classification (ten ranks in the example presented earlier) and up to twenty observations at each level. The program also accomodates an unequal number of observations at each level. The two-way classification ANOVA procedure allows ten levels of each classification and up to ten replications. The three-way classification again allows up to ten levels of each of the three classifications but due to computer memory limitations does not allow for replications. The two- and three-way classification procedures calculate interaction statistics as well as the main classification statistics. All three procedures are accurate to a minimum of three decimal places.

IV. CONCLUSION

When microcomputers were first introduced only a few years ago, they were widely perceived as interesting toys used primarily for video games. This perception is rapidly changing and today microcomputers are increasingly accepted as serious tools for many small business applications. Just as the military services have adopted electric typewriters, photo-copy machines and other modern office machinery, it is not hard to imagine that relatively inexpensive microcomputers will also be adopted and find their way to all levels of command. A popular commercial application of microcomputers is in the area of wordprocessing. It may well be that this will be the initial application of microcomputers in the military services. Whatever the intended application might be, the end result will be ready access to a high degree of computational power.

If this potential ready access to computational power is to be put to use by the military officer trained in specialized areas such as operations research, then it is up to these officers to develop the specialized software. Specialized software for microcomputers has been the subject of this thesis. It has presented a software package containing some of the basic statistical analysis tools required for effective analysis work. Many areas of interest to the trained analyst have not been addressed. Other recent efforts in this direction are addressed by Morgeson [Ref. 26] who developed a similar package for the Apple II Plus microcomputer and Duff [Ref. 27] who has developed a network optimization package for the same microcomputer. Additional efforts in the development of statistical analysis software are highly recommended and encouraged.

APPENDIX A

ACCURACY COMPARISONS FOR PROBABILITIES

NORMAL DISTRIBUTION

	PROBABILITY	
STATISTIC	APPROXIMATION	TABLED VALUE
0.00	.5000	.5000
0.25	.5987	.5987
0.50	.6915	.6915
0.75	.7734	.7734
1.00	.8413	.8413
1.25	.8944	.8944
1.50	.9332	.9332
1.75	.9599	.9599
2.00	.9772	.9772
2.25	.9878	.9878
2.50	.9938	.9938
2.75	.9970	.9970
3.00	.9987	.9987
3.25	.9994	.9994
3.50	.9998	.9998
3.75	.9999	.9999
4.00	1.0000	1.0000

CHI-SQUARE DISTRIBUTION

		PROBABILITY	
STATISTIC	D.F.	APPROXIMATION	TABLED VALUE
7.88	1	.9950	.9950
5.99	2	.9500	.9500
1.21	3	.2495	.2500
3.36	4	.5005	.5000
9.24	5	.9001	.9000
2.18	8	.0250	.0250
3.57	12	.0100	.0100
27.5	15	.9751	.9750
34.8	18	.9900	.9900
10.1	19	.0496	.0500
9.59	20	.0250	.0250
33.9	22	.9497	.9500
34.4	25	.9004	.9000
27.3	28	.4990	.5000
49.6	29	.9900	.9900
47.0	30	.9751	.9750

STUDENT's - t DISTRIBUTION

STATISTIC	D.F.	PROBABILITY APPROXIMATION	PROBABILITY TABLED VALUE
31.821	1	.9900	.9900
1.886	2	.9000	.9000
.271	4	.6001	.6000
3.365	5	.9900	.9900
1.943	6	.9500	.9500
.703	9	.7501	.7500
2.228	10	.9750	.9750
4.140	14	.9995	.9995
1.341	15	.9001	.9000
.257	19	.6000	.6000
2.086	20	.9750	.9750
2.492	24	.9900	.9900
3.725	25	.9995	.9995
1.311	29	.8999	.9000
2.750	30	.9949	.9950
1.684	40	.9500	.9500
1.296	60	.9000	.9000
3.373	120	.9995	.9995

F - DISTRIBUTION

STATISTIC	D.F./D.F.	PROBABILITY APPROXIMATION	TABLED VALUE
647.80	1/1	.9750	.9750
198.50	1/2	.9950	.9950
53.59	3/1	.9000	.9000
5.79	2/5	.9501	.9500
21.92	4/6	.9990	.9990
4.24	5/10	.9751	.9750
5.76	6/12	.9950	.9950
1.93	8/25	.9001	.9000
2.16	10/30	.9495	.9500
3.64	12/40	.9990	.9990
2.06	15/60	.9749	.9750
2.09	24/120	.9950	.9950
5.23	10/3	.9000	.9000
46.76	15/4	.9990	.9990
5.17	20/6	.9750	.9750
2.63	30/30	.9950	.9950
63.06	120/1	.9000	.9000

APPENDIX B

ACCURACY COMPARISIONS FOR PERCENTILES

INVERSE NORMAL D.˜TRIBUTION

	PERCENTILE	
PROBABILITY	APPROXIMATION	TABLED VALUE
.0005	-3.29	-3.29
.0102	-2.32	-2.32
.2514	-0.67	-0.67
.4960	-0.01	-0.01
.5199	0.05	0.05
.6554	0.40	0.40
.7549	0.69	0.69
.9015	1.29	1.29
.9251	1.44	1.44
.9750	1.96	1.96
.9901	2.33	2.33
.9951	2.58	2.58
.9990	3.09	3.09
.9995	3.29	3.29

INVERSE CHI-SQUARE DISTRIBUTION

PROBABILITY	D.F.	PERCENTILE APPROXIMATION	PERCENTILE TABLED VALUE
.500	1	.455	.455
.950	2	5.99	5.99
.025	3	.216	.216
.990	5	15.1	15.1
.005	8	1.34	1.34
.975	10	20.5	20.5
.010	12	3.57	3.57
.900	15	22.3	22.3
.995	20	40.0	40.0
.750	22	26.0	26.0
.025	25	13.1	13.1
.975	28	44.5	44.5
.100	30	20.6	20.6

INVERSE STUDENT's - t DISTRIBUTION

		PERCENTILE	
PROBABILITY	D.F.	APPROXIMATION	TABLED VALUE
.9900	1	31.840	31.821
.9500	2	2.920	2.920
.0250	3	-3.182	-3.182
.7500	5	.726	.727
.9750	10	2.229	2.228
.9995	15	4.073	4.073
.0500	18	-1.734	-1.734
.9000	20	1.326	1.325
.6000	25	.256	.256
.2500	28	-.683	-.683
.9995	30	3.646	3.646
.0100	40	-2.424	-2.423
.0005	60	-3.460	-3.460
.9950	120	2.618	2.617

INVERSE F - DISTRIBUTION

PROBABILITY	D.F./D.F.	PERCENTILE APPROXIMATION	PERCENTILE TABLED VALUE
.975	1/1	647.50	647.80
.995	1/2	198.10	198.50
.900	3/1	53.61	53.59
.950	2/5	5.79	5.79
.990	4/6	9.15	9.15
.975	5/10	4.24	4.24
.995	6/12	5.76	5.76
.900	8/25	1.93	1.93
.950	10/30	2.17	2.16
.990	12/40	2.67	2.66
.975	15/60	2.06	2.06
.995	24/120	2.09	2.09
.900	10/3	5.23	5.23
.975	60/20	2.23	2.22
.995	30/30	2.63	2.63
.900	120/1	63.34	63.06
.950	120/120	1.35	1.35

COMPUTER LISTINGS

PROGRAM #1

CONTROL MODULE / INSTRUCTIONS / PROGRAM SELECTION MENU

```
10 REM *** INSTRUCTIONS/PROGRAM SELECTION SUBROUTINE
20 IF K9<>0 GOTO 1000ELSE CLEAR
30 RANDOM
40 CLEAR 350
50 DIM X(201),Y(201),NI(15)
60 DEFINT I-K : DEFSTR Z
65 LB$=CHR$(123) : RB$=CHR$(125) : MB$=CHR$(176) : BL$=CHR$(252)
70 REM *** INSTRUCTIONS ***
80 CLS : PRINT TAB(20);"STATISTICAL ANALYSIS" : PRINT TAB(23);"by R. P. Isbell" : PRINT TAB(11);"Naval Po
stgraduate School, Monterey, Ca."
90 IF K9<>0 GOTO 290
100 PRINT : PRINT"    This program is the  first  of six 'chained ' programs in"
110 PRINT"this package. It contains the first set of statistical analy-"
120 PRINT"sis  subroutines and also serves as the control module for the"
130 PRINT"package. This program must always be loaded first as it accom-"
140 PRINT"plishes variable definition, dimensions arrays and, most impor-"
150 PRINT"tantly,  positions the variable pointer  to  allow  subsequent "
160 PRINT"programs  to  be loaded without loss of variables.  If at any "
170 PRINT"time it becomes necessary to interupt program execution and re-"
180 PRINT"issue the RUN command, the action must be taken with Program #1"
190 PRINT"(i.e. this program) loaded."
200 PRINT : PRINT"    Press the SPACE BAR to display additional information.";
210 IF INKEY$ <> " " GOTO 210
220 CLS : PRINT"    Data input for programs #1 thru #4 is accomplished in pro- gram #1 and is limited to a data
vector of  200  observations.  All data and variables are saved as you switch back and forth  between the first fo
ur programs."
230 PRINT:PRINT"    When either program #5 or program #6 is loaded all current variables are cleared in order t
o provide sufficient memory for the multidimensional data arrays used by these programs.  Any"
240 PRINT"data which was entered in program  #1  that  you want to retain should be saved to tape prior to loadi
ng either of the last two programs.
250 PRINT @896,"PRESS SPACE BAR TO DISPLAY PROGRAM SELECTION MENU:"
260 IF INKEY$<>" " GOTO 260
270 K9=1 : REM *** PROGRAM SELECTION MENU ***
280 CLS : PRINT TAB(20);"PROGRAM SELECTION MENU"
290 PRINT @128,"#1  Data I/O, Transformations, Prob. & Inv. Prob. Distributions":PRINT
300 PRINT"#2  Sample Stat's, 7-Num-Sum, Data Plots":PRINT
310 PRINT"#3  Parametric Tests of Hypotheses & Confidence Intervals":PRINT
320 PRINT"#4  Robust Estimates and Confidence Intervals":PRINT
330 PRINT"#5  Multiple Linear Regression":PRINT
340 PRINT"#6  Analysis of Variance"
350 KP=896 : KS=6 : GOSUB 500
420 IF IO=1 GOTO 1000
430 IF IO=2 THEN ZT="1" : GOTO 450
440 IF IO=3 OR IO=4 THEN ZT="2" ELSE ZT="3"
450 IF IO=3 OR IO=5 THEN IO=1 ELSE IO=2
460 CLS : PRINT"INSERT TAPE #";ZT;" INTO ESF-80 AND PRESS SPACE BAR"
470 IF INKEY$<>" " GOTO 470
480 @LOAD IO
500 REM *** OPTION SELECTION SUBROUTINE ***
510 PRINT @KP+5,"OPTION DESIRED ";LB$;MB$;RB$;
520 ZI=INKEY$ : IF ZI="" THEN 520ELSE IO=VAL(ZI)
```

```
530 IF IO<1 OR IO>KS THEN PRINT @KP, BL$;: PRINT @KP,"***  NOT A VALID OPTION  ***";: FOR I=1 TO 100
0 : NEXT : PRINT @KP, BL$; : GOTO 510
540 RETURN
600 PRINT @KP+5,LB$;" ";L$;" or ";R$;" ";RB$;"  ";LB$;MB$;RB$;
610 ZQ=INKEY$ : IF ZQ="" THEN 610ELSE PRINT @KP+18,ZQ;: FOR I=1 TO 100 : NEXT
620 IF ZQ<>L$ AND ZQ<>CHR$(ASC(L$)-32) AND ZQ<>R$ AND ZQ<>CHR$(ASC(R$)-32) THEN 600
630 RETURN
```

PROGRAM #1 MENU

```
1000 REM  ***  P R O G R A M  # 1  ***
1010 REM  ***  M A I N  M E N U  ***
1020 CLS : PRINT TAB(20);"***  MENU  ***"
1030 PRINT @128,"(1) Data I/O
1040 PRINT"(2) Data Transformations
1050 PRINT"(3) Probability Distributions
1060 PRINT"(4) Inverse Probability Distributions
1070 PRINT"(5) Return to Program Selection Menu
1080 KP=704 : KS=5 : GOSUB 500
1150 IF I4=1 AND IO=2 OR I4=0 AND IN=0 AND IO=2 PRINT@704,CHR$(253);:PRINT @704,"THERE IS NO DATA T
O TRANSFORM";:FOR I=1 TO 1500:NEXT:PRINT @704,CHR$(253);:GOTO 1080
1160 ON IO GOSUB 2000, 3500, 5000, 5200, 270
1170 GOTO 1020
```

DATA INPUT / OUTPUT

```
2000 REM  ***  D A T A  I N P U T / O U T P U T  S U B R O U T I N E
2010 CLS : PRINT @11,"*** D A T A  I N P U T / O U T P U T ***
2020 PRINT:PRINT:PRINT"MENU:"
2030 PRINT"  (1) Input Data via Keyboard
2040 PRINT"  (2) Input Data via Tape
2050 PRINT"  (3) Input Summary Statistics ONLY
2060 PRINT"  (4) Create a random sample using random number generator" : PRINT"      (for demonstration purp
oses)
2070 PRINT"  (5) SAVE data to tape
2080 KP=768 : KS=5 : GOSUB 500
2100 IF IO=1 OR IO=2 THEN I5=1 ELSE I5=0
2120 ON IO GOTO 2200, 2700, 2900, 4000, 3300
```

INPUT DATA VIA KEYBOARD

```
2200 REM  ***  I N P U T  D A T A  V I A  K E Y B O A R D  ***
2210 CLS : I4=0
2220 PRINT"To end data input just press ENTER in response to prompt"
2230 IN=0
2240 PRINT
2250 X$="" : PRINT"Enter X(";IN+1;")";:INPUT X$
2260 IF X$="" GOTO 2310
2270 T$=LEFT$(X$,1) : IF T$="0" OR T$="1" OR T$="2" OR T$="3" OR T$="4" OR T$="5" OR T$="6" OR T$="7" OR
 T$="8" OR T$="9" OR T$="." OR T$="-" OR T$="+" THEN GOTO 2290
2280 PRINT"Reenter last value" : GOTO 2250
2290 IN=IN+1 : X(IN)=VAL(X$) : Y(IN)=X(IN)
2295 IF IN=1 THEN 2250 ELSE IF X(IN)<X(IN-1) THEN I5=0
2298 IF IN>199 THEN 2310
2300 GOTO 2250
2310 PRINT: IF IN=0 THEN PRINT"NO DATA ENTERED"; ELSE IF IN=1 PRINT"ANALYSIS CAN NOT BE PERFORM
ED ON A SAMPLE OF 1"; ELSE PRINT"Number of data points =";IN : FOR I=1 TO 500 : NEXT
2320 FOR I=IN+1 TO 200 : X(I)=0 : Y(I)=X(I) : NEXT
2330 IF IN=0 THEN RETURN ELSE IF IN=1 THEN IN=0 : RETURN
2340 IF I5=1 GOTO 3800
```

SHELL SORT

```
2400 REM *** S H E L L  S O R T  R O U T I N E ***
2410 CLS : PRINT @458,"***  O R D E R I N G   D A T A ***"
2420 IF IN>3 GOTO 2510
2430 IF IN=3 GOTO 2460
2440 IF X(1)>X(2) THEN XT=X(1) : X(1)=X(2) : X(2)=XT
2450 GOTO 2500
2460 IF X(1)>X(2) THEN XT=X(2) : X(2)=X(1) : X(1)=XT
2470 IF X(2)>X(3) THEN XT=X(3) : X(3)=X(2) : X(2)=XT
2480 IF X(1)>X(2) THEN XT=X(2) : X(2)=X(1) : X(1)=XT
2500 FOR I=1 TO IN : Y(I)=X(I) : NEXT I : GOTO 2660
2510 IM=INT(LOG(IN)/LOG(2))-1
2520 H = 2[IM - 1
2530 K = H
2540 FOR I = K + 1  TO IN
2550 XT=X(I)
2560 J = I - K
2570  IF J < 1 GOTO 2620
2580  IF XT >= X(J) GOTO 2620
2590 X(J+K) = X(J) : Y(J+K) = Y(J)
2600 J = J - K
2610 GOTO 2570
2620 X(J+K) = XT : Y(J+K) = XT
2630 NEXT I
2640 H = (H - 1)/2
2650 IF H >= 1 GOTO 2530
2660 GOSUB 3800
2670 RETURN
```

INPUT DATA VIA TAPE

```
2700 REM *** I N P U T  D A T A  V I A  T A P E ***
2710 CLS : I4=0 : PRINT"Insert DATA TAPE into ESF-80"
2720 PRINT : PRINT"Press SPACE BAR after inserting the tape"
2730 IF INKEY$ <> " " GOTO 2730
2740 PRINT : INPUT "Enter Data File #";FI
2750 @OPEN FI
2760 I=1
2770 @INPUT X(I)
2780 Y(I)=X(I)
2785 IF I=1 THEN 2800 ELSE IF X(I)<X(I-1) THEN I5=0
2790 IF X(I)=999999 GOTO 2810
2800 I=I+1 : GOTO 2770
2810 @CLOSE
2820 IN=I-1 : PRINT : PRINT"NUMBER OF DATA POINTS =";I-1 : FOR I=1 TO 1500 : NEXT I : IF I5=0 THEN GOT
O 2410ELSE GOTO 3800
```

INPUT SUMMARY STATISTICS ONLY

```
2900 REM *** I N P U T  S U M M A R Y  S T A T I S T I C S  O N L Y ***
2910 I4=1 : CLS : PRINT TAB(20)"SUMMARY STATISTICS INPUT"
2920 PRINT : INPUT"Sample Size";IN : IF IN=1 OR IN=0 GOTO 2310
2930 PRINT @192,"Is sample assumed to be from an Exponential Distribution?"
2940 KP=256 : L$="y" : R$="n" : GOSUB 600
2950 IF ZQ="n" OR ZQ="N" GOTO 2990ELSE PRINT @192,"Will you enter ";LB$;"M";RB$;"ean  or  Exponential Para
meter  ";LB$;"L";RB$;"ambda";
2960 KP=256 : L$="m" : R$="l" : GOSUB 600
2970 IF ZQ="m" OR ZQ="M" THEN PRINT : INPUT"Enter Mean";XB : V1=XB*XB : V=V1 : GOTO 3170
2980 PRINT : INPUT"Enter Lambda";XB : XB=1/XB : V1=XB*XB : V=V1 : GOTO 3170
2990 PRINT @192, "Is the sample assumed to be from a Bernoulli Population?"
3000 KP=256 : L$="y" : R$="n" : GOSUB 600
```

```
3010 IF ZQ="n" OR ZQ="N" THEN GOTO 3030ELSE PRINT : PRINT"Enter number of occurrences out of";IN;"trials";:INPUT XB : IF XB>IN OR XB<0 GOTO 3020ELSE IF XB<>INT(XB) GOTO 3020ELSE XB=XB/IN : V1=XB*(1-XB) : V=V1 : GOTO 3170
3020 PRINT @192,CHR$(250) : PRINT @192,"YOU MUST ENTER AN INTEGER FROM 0 TO";IN : PRINT: FOR I=1 TO 2000 : NEXT : GOTO 3010
3030 PRINT @192,BL$ : PRINT : PRINT @192,"";:INPUT"Sample Mean";XB
3040 PRINT : PRINT"Will you enter variance or std. deviation ?"
3050 KP=389 : L$="v" : R$="d" : GOSUB 600
3060 IF ZQ="v" OR ZQ="V" THEN V$="Sample Variance" ELSE V$="Sample Std. Deviation"
3070 PRINT : PRINT V$;:INPUT SV
3080 IF V$="Sample Variance" THEN GOTO 3100ELSE V1=SV*SV : V=((IN-1)/IN)*V1
3090 GOTO 3170
3100 PRINT"Is the sample variance:   ";CHR$(123);"1";CHR$(125);" the M.L.E. (i.e. 1/n)
3110 PRINT"                           ";CHR$(123);"2";CHR$(125);" unbiased estimator (i.e. 1/n-1)
3120 PRINT @640,"SELECT APPROPRIATE ESTIMATOR ";CHR$(123);" ";CHR$(125)
3130 ZI=INKEY$ : IF ZI="" GOTO 3130ELSE PRINT @670,ZI:FOR I=1 TO 100: NEXT
3140 IF ZI<>"1" AND ZI<>"2" THEN PRINT @670," ";:GOTO 3130
3150 IF ZI="1" V=SV : V1=(IN/(IN-1))*SV
3160 IF ZI="2" THEN V1=SV : V=((IN-1)/IN)*SV
3170 CLS : PRINT"When summary statistics are entered in lieu of data, several    subroutines are not available. These include:"
3180 Z1=CHR$(123):Z2=CHR$(125):PRINT:PRINT"    ";Z1;" Data Transformations ";Z2 : PRINT"    ";Z1;" Five Number Summary ";Z2 : PRINT"    ";Z1;" All Data Plots ";Z2 : PRINT"    ";Z1;" All of the Subroutines in Prog. #4 ";Z2
3190 FOR I=1 TO 200 : X(I)=0 : Y(I)=0 : NEXT
3200 PRINT @896,"PRESS SPACE BAR TO CONTINUE"
3210 IF INKEY$<>" " GOTO 3210
3220 RETURN
```

SAVE DATA TO TAPE

```
3300 REM *** S A V E   D A T A   T O   T A P E ***
3310 CLS : PRINT"INSTRUCTIONS:"
3320 PRINT"  (1) Insert DATA TAPE
3330 PRINT"  (2) If you use a blank tape then FILE # will be 1
3340 PRINT"  (3) If the tape is not blank then the FILE # must be
3350 PRINT"      the number of the last data file recorded PLUS 1
3360 PRINT : PRINT "When the tape is ready press SPACE BAR
3370 IF INKEY$ <> " " GOTO 3370
3380 PRINT : INPUT"Enter FILE #";IO
3390 Y(IN+1)=999999
3400 @OPEN IO
3410 FOR I=1 TO IN+1
3420 @PRINT Y(I)
3430 NEXT I
3440 @CLOSE
3450 PRINT IN;"data points recorded as DATA FILE #";IO : FOR I=1 TO 2000 : NEXT : RETURN
```

DATA TRANSFORMATIONS

```
3500 REM *** D A T A   T R A N S F O R M A T I O N   S U B R O U T I N E
3510 CLS : PRINT TAB(22)"DATA TRANSFORMATIONS"
3520 PRINT : PRINT "MENU:" : PRINT"(1) NO Transformation
3530 PRINT"(2) Square Root Transformation
3540 PRINT"(3) Cube Root Transformation
3550 PRINT"(4) Logarithmic Transformation
3560 PRINT : PRINT"  NOTE 1: Other transformations must be made before data entry
3570 PRINT"  NOTE 2: Raw data may be recovered by returning to this sub-
3580 PRINT"          routine and selecting Option 1
3590 PRINT"  NOTE 3: Option 2 will not work with negative data and          option 4 requires all data greater than zero
```

```
3600 KP=896 : KS=4 : GOSUB 500
3610 CLS : PRINT"EXECUTING TRANSFORMATION OPTION";IO
3620 FOR I=1 TO IN
3630 ON IO GOTO 3640, 3650, 3670, 3690
3640 Y(I)=X(I) : GOTO 3710
3650 IF X(I) < 0 CLS : PRINT"CAN NOT TAKE THE SQUARE ROOT OF NEGATIVE DATA!" : FOR J=1 TO 2000 : NE
XT J : GOTO 3500
3660 Y(I)=SQR(X(I)) : GOTO 3710
3670 Y(I)=ABS(X(I))[.3333333 : IF X(I) < 0 Y(I)=Y(I)*-1
3680 GOTO 3710
3690 IF X(I) <= 0 CLS : PRINT"CAN NOT TAKE THE LOG OF DATA LESS THAN OR EQUAL TO ZERO" : FOR J=1 T
O 2000 : NEXT J : GOTO 3500
3700 Y(I)=LOG(X(I))
3710 NEXT I
```

CALCULATE SAMPLE STATISTICS

```
3800 REM *** C A L C U L A T E   S A M P L E   S T A T I S T I C S ***
3810 CLS : PRINT @454,"* I N I T I A L I Z I N G   D A T A   B A S E *"
3820 SM=0 : SUM=0 : S2=0 : FOR I=1 TO IN : SM=SM+Y(I) : DV=Y(I)-SUM : SUM=SUM+DV/I : S2=S2+DV * (Y(I)-SUM
) : NEXT
3830 XB=SM/IN : V1=S2/(IN-1) : V=S2/IN
3840 G1=0 : G2=0 : FOR I=1 TO IN : G1=G1+(Y(I)-XB)[3 : G2=G2+(Y(I)-XB)[4 : NEXT I : G1=G1/IN/V[1.5 : G2=(G2
/IN/V[2)-3
3850 RETURN
```

RANDOM NUMBER GENERATOR

```
4000 REM *** R A N D O M   N U M B E R   G E N E R A T O R   S U B R O U T I N E
4010 CLS : I4=0 : PRINT@18,"RANDOM NUMBER GENERATOR"
4020 PRINT : PRINT : PRINT"Distribution Options:"
4030 PRINT"  (1) Normal
4040 PRINT"  (2) Uniform
4050 PRINT"  (3) Exponential
4060 PRINT"  (4) Bernoulli
4070 KP=645 : KS=4 : GOSUB 500
4080 PRINT@704,""; : INPUT"How many random numbers do you want";IN
4085 IF IN<=1 THEN PRINT @640, CHR$(250) : PRINT @640,"WHAT? ... Please request a sample size of at least 2
" : GOTO 4080
4090 ON IO GOSUB 4300, 4400, 4600, 4700
4100 CLS : PRINT"Do you want to see the random sample ";: L$="y" : R$="n" : KP=35 : GOSUB 600
4110 IF ZQ="n" OR ZQ="N" THEN 2410ELSE PRINT
4120 J=0
4130 FOR I=1 TO IN
4140 J=J+1 : IF J > 5 PRINT" "
4150 IF J > 5 THEN J=1
4160 PRINT USING"#######.###";X(I);:PRINT" ";
4170 IF INT(I/50)=I/50 GOTO 4190
4180 GOTO 4210
4190 PRINT : PRINT"PRESS SPACE BAR TO DISPLAY NEXT SET OF NUMBERS"
4200 IF INKEY$ <> " " GOTO 4200
4210 NEXT I
4220 PRINT : PRINT"Press SPACE BAR to continue"
4230 IF INKEY$ <> " " GOTO 4230
4240 GOTO 2410
```

NORMAL RNG

```
4300 REM *** N O R M A L   R N G ***
4310 CLS : PRINT @10,"NORMAL RANDOM NUMBERS BY BOX-MULLER METHOD"
4320 PRINT : PRINT
4330 INPUT"What is the mean";MU
4340 INPUT"What is the variance";VAR : SIGMA=SQR(VAR)
4350 FOR I=1 TO IN STEP 2
```

```
4360 U1=RND(0) : U2=RND(0)
4370 X(I)=SQR(-2*LOG(U1))*COS(6.28315*U2):X(I)=X(I)*SIGMA + MU : Y(I)=X(I)
4380 X(I+1)=SQR(-2*LOG(U1))*SIN(6.28315*U2):X(I+1)=X(I+1)*SIGMA + MU : Y(I+1)=X(I+1)
4390 NEXT I
4395 RETURN
```

UNIFORM RNG

```
4400 REM *** U N I F O R M  R N G ***
4410 CLS
4420 PRINT@20,"UNIFORM RANDOM NUMBERS"
4430 PRINT : PRINT : PRINT"   (1) Continuous
4440 PRINT"   (2) Uniform Integers
4450 KP=384 : KS=2 : GOSUB 500
4460 PRINT : INPUT"Enter Lower Limit";LL
4470 INPUT"Enter Upper Limit";UL
4480 ON IO GOTO 4490, 4540
4490 FOR I=1 TO IN
4500 X(I)=RND(0)*(UL-LL) + LL
4510 Y(I)=X(I)
4520 NEXT I
4530 RETURN
4540 FOR I=1 TO IN
4550 X(I)=INT(RND(0)*(UL-LL) + LL ) : IF X(I) <> LL THEN  X(I)=X(I)+1
4560 Y(I)=X(I)
4570 NEXT I
4580 RETURN
```

EXPONENTIAL RNG

```
4600 REM *** E X P O N E N T I A L  R N G ***
4605 CLS : PRINT @15,"EXPONENTIAL RANDOM NUMBERS"
4610 PRINT @128,"Will you enter the Exponential Parameter Lambda or the Mean" : KP=192 : L$="l" : R$="m" : GO
SUB 600
4620 IF ZQ="l" OR ZQ="L" THEN ZP="Lambda" ELSE ZP="Mean"
4630 PRINT @256,"Enter ";ZP;:INPUT MU
4640 IF ZP="Lambda" THEN MU=1/MU
4650 FOR I=1 TO IN
4660 U1=RND(0)
4670 X(I)= -MU * LOG(U1)
4680 Y(I)=X(I)
4690 NEXT I
4695 RETURN
```

BERNOULLI RNG

```
4700 REM *** B E R N O U L L I  R N G ***
4710 CLS
4720 PRINT@15,"BERNOULLI RANDOM NUMBERS
4730 PRINT : INPUT"What is the probability of success on a single trial";PS
4740 FOR I=1 TO IN
4750 X(I)=RND(0)
4760 IF X(I) <= PS THEN X(I)=1 ELSE X(I)=0
4770 NEXT I
4780 RETURN
```

PROBABILITY DISTRIBUTIONS MENU

```
5000 REM *** P R O B A B I L I T Y  D I S T R I B U T I O N S  S U B R O U T I N E
5010 CLS : PRINT TAB(18)"PROBABILITY DISTRIBUTIONS
5020 PRINT:PRINT: I2=0
5030 PRINT"MENU:"
5040 PRINT"(1)  Normal Distribution
5050 PRINT"(2)  Binomial Distribution
5060 PRINT"(3)  Negative Binomial Distribution
5070 PRINT"(4)  Poisson Distribution
5080 PRINT"(5)  Exponential Distribution
5090 PRINT"(6)  Chi-Square Distribution
5100 PRINT"(7)  Student's - t  Distribution
5110 PRINT"(8)  F-Distribution
5120 KP=896 : KS=8 : GOSUB 500
5130 ON IO GOSUB 6000,7000,8000,9000,10000,11000,12000,13000
5140 IF IO=2 OR IO=3 OR IO=4 OR IO=5 THEN RETURN
5150 PRINT : PRINT : PRINT "Prob. ( X <=";PX;") = "; : PRINT USING"#.####";FOX
5160 PRINT "Prob. ( X > ";PX;") = "; : PRINT USING"#.####";QOX
5170 PRINT @960,"  Run Again or Quit?";: KP=978 : L$="r" : R$="q" : GOSUB 600
5180 CLS : I2=0 : IF ZQ="r" OR ZQ="R" GOTO 5130ELSE RETURN
5190 GOTO 1020
```

INVERSE DISTRIBUTIONS MENU

```
5200 REM *** I N V E R S E  D I S T R I B U T I O N S ***
5210 CLS : PRINT TAB(15)"INVERSE PROBABILITY DISTRIBUTIONS
5220 PRINT @192,"MENU:":PRINT"(1)  Inverse Normal
5230 PRINT"(2)  Inverse Chi-Square
5240 PRINT"(3)  Inverse Student's - t
5250 PRINT"(4)  Inverse F
5260 PRINT"(5)  Inverse Exponential"
5270 KP=960 : KS=5 : GOSUB 500
5280 I2=0
5290 ON IO GOSUB 14000, 15000, 16000, 17000, 18000
5300 IF IO=5 GOTO 5350
5310 PRINT : PRINT "For  F(x) = Prob. ( X <= x ) =";PR
5320 IF XQ=9E36 THEN PRINT "x =  infinity" ELSE IF XQ=9E-36 PRINT "x =  - infinity"
5330 IF XQ=9E36 OR XQ=9E-36 THEN GOTO 5350
5340 PRINT "x = "; : IF ABS(XQ)>1000 THEN PRINT INT(XQ) ELSE IF ABS(XQ)>100 PRINT USING"####.#";XQ EL
SE IF ABS(XQ)>10 PRINT USING"###.##";XQ ELSE PRINT USING"##.###";XQ
5350 PRINT @960,"  Run Again or Quit";: L$="r" : R$="q" : KP=976 : GOSUB 600
5360 IF ZQ="R" OR ZQ="r" GOTO 5280ELSE RETURN
```

NORMAL DISTRIBUTION

```
6000 REM  *** N O R M A L  D I S T R I B U T I O N  S U B R O U T I N E
6010 IF I2=1 GOTO 6070
6020 CLS:PRINT @20,"NORMAL DISTRIBUTION"
6030 PRINT:PRINT:INPUT"Enter Mean";MU
6040 INPUT"Enter Variance";VAR
6050 IF VAR<=0 PRINT"** ERROR **   ";VAR"IS NOT A VALID VARIANCE":GOTO 6040
6060 INPUT"Enter Normal Statistic  x ";NX : IF I2=0 THEN PX=NX
6070 SD=SQR(VAR)
6080 XN=(NX-MU)/SD:AX=ABS(XN)
6090 XZ=.3989423*EXP((-AX*AX)/2)
6100 NT=1/(1+.2316419*AX)
6110 QOX=XZ*((.3193815*NT)-(.3565638*NT[2)+(1.781478*NT[3)-(1.821256*NT[4)+(1.330274*NT[5))
6120 IF XN<0 GOTO 6150
6130 FOX=1-QOX
6140 RETURN
6150 FOX=QOX
```

```
6160 QOX=1-FOX
6170 RETURN
```

BINOMIAL DISTRIBUTION

```
7000 REM *** B I N O M I A L  D I S T R I B U T I O N  S U B R O U T I N E ***
7010 IF I2=1 GOTO 7330
7020 CLS : PRINT TAB(20)"BINOMIAL DISTRIBUTION
7030 PRINT
7040 PRINT"    The normal scenario for the Binomial Distribution is K"
7050 PRINT"occurrences out of  N  trials where  P  is the probability "
7060 PRINT"of occurrence on a single trial.
7070 PRINT : INPUT"Enter N";N
7080 IF N<0 OR N<>INT(N) THEN PRINT"** ERROR **   N  MUST BE A POSITIVE INTEGER":GOTO 7070
7090 INPUT"Enter P";PB
7100 IF PB=1 THEN PRINT"THE PROBABILITY OF";N;"OUT OF";N;"= 1.0":GOTO 5170
7110 IF PB=0 THEN PRINT"THE PROBABILITY OF 0 OUT OF";N;"= 1.0":GOTO 5170
7120 IF PB<0 OR PB>1 THEN PRINT"** ERROR **  ";PB;"IS NOT A VALID PROBABILITY":GOTO 7090
7130 INPUT"Enter K";XK
7140 K=XK
7150 IF K<0 OR K<>XK THEN PRINT"** ERROR **   K  MUST BE A POSITIVE INTEGER":GOTO 7130
7160 IF K>N THEN PRINT"THE PROBABILITY OF ";K;" OUT OF ";N;" = 0":GOTO 5170
7170 IF K=0 OR N=K THEN BC=1 : GOTO 7240
7180 IF K<0 THEN PR=0 : GOTO 7250
7190 TB=N : BB=K
7200 IF N<120 GOTO 7230
7210 GOSUB 7740
7220 GOTO 7250
7230 GOSUB 7430
7240 PR = BC * PB[K * (1-PB)[(N-K)
7250 PRINT : PRINT"Prob. of exactly";K;"occurrences out of";N;"trials = ";:PRINT USING"#.####";PR
7260 PRINT @960,"Do you want Cumulative Probability ";: KP=992 : L$="y" : R$="n" : GOSUB 600
7270 IF ZQ="n" OR ZQ="N" PRINT @960,BL$;: GOTO 5170
7280 PRINT : INPUT"Enter K";XK
7290 K=XK : IF K>N THEN GOTO 7160ELSE IF K<0 OR K<>XK PRINT"** ERROR **   K  MUST BE A POSITIVE IN
TEGER":GOTO 7280
7300 IF N<=45 GOTO 7330
7310 IF PB>.1 AND PB<.9 AND  PB*N>5 THEN PRINT"NORMAL APPROXIMATION WILL BE USED" :GOSUB 7580:
 GOTO 7410
7320 IF PB<=.1 OR PB>=.9 THEN PRINT"POISSON APROXIMATION WILL BE USED" : GOSUB 7660: GOTO 7410
7330 PR=0
7340 IF K<0 GOTO 7410
7350 IF K >= N THEN PR=1 : GOTO 7410
7360 FOR BB=0 TO K
7370 TB=N : GOSUB 7430
7380 PR=PR+ BC * PB[BB * (1-PB)[(N-BB)
7390 IF PR>=1 GOTO 7410
7400 NEXT BB
7410 IF I2=1 RETURN ELSE PRINT : PRINT"Prob. of";K;"or less occurrences out of";N;"trials = ";:PRINT USING"
#.###";PR
7420 PRINT : GOTO 7260
7430 BC=0 : B=0 : BT=0 : SB=BB : QB=TB-BB
7440 FOR BI=1 TO TB
7450 IF SB>=QB GOTO 7510
7460 IF SB >= BI THEN B=B + LOG(BI)
7470 IF QB > BI GOTO 7550
7480 IF QB = BI THEN BC=-B : GOTO 7550
7490 BC=BC + LOG(BI)
7500 GOTO 7550
7510 IF QB >= BI THEN B=B + LOG(BI)
7520 IF SB > BI THEN GOTO 7550
7530 IF SB = BI THEN BC=-B : GOTO 7550
```

```
7540 BC=BC + LOG(BI)
7550 NEXT BI
7560 BC=EXP(BC)
7570 RETURN
7580 MU=PB
7590 VAR=(PB * (1-PB) )/N
7600 NX=(K + .5)/N
7610 I2=1
7620 GOSUB 6000
7630 I2=0
7640 PR=FOX
7650 RETURN
7660 IF PB<=.1 THEN LAM=N*PB ELSE LAM=N*(1-PB)
7670 TI=1
7680 IF PB<=.1 THEN NC=K ELSE NC=N-K-1
7690 I2=1
7700 GOSUB 9230
7710 I2=0
7720 IF PB>=.9 THEN PR=1-PR
7730 RETURN
7740 REM  * * *  NORMAL APPROXIMATION
7750 IF PB>=.9 AND (1-PB)*N<5 THEN GOTO 7880
7760 IF PB<.1 AND PB*N<5 THEN GOTO 7880
7770 MU=PB
7780 VAR=(PB * (1-PB))/N
7790 NX=(K+.5)/N
7800 I2=1
7810 GOSUB 6000
7820 PR=FOX
7830 NX=(K-.5)/N
7840 GOSUB 6000
7850 PR=PR-FOX
7860 I2=0
7870 RETURN
7880 REM  *** POISSON APPROXIMATION
7890 IF PB<=.1 THEN LAM=N*PB ELSE LAM=N*(1-PB)
7900 TI=1
7910 IF PB<=.1 THEN NC=K ELSE NC=N-K
7920 I2=1
7930 GOSUB 9000
7940 I2=0
7950 IF PR>=.9 THEN PR=1-PR
7960 RETURN
```

__NEGATIVE BINOMIAL DISTRIBUTION__

```
8000 REM * * *  N E G A T I V E   B I N O M I A L   S U B R O U T I N E  * * *
8010 IF I2=1 GOTO 8170
8020 CLS : PRINT TAB(15)"NEGATIVE BINOMIAL DISTRIBUTION
8030 PRINT : PRINT"  The normal scenario for the Negative Binomial Distribution is"
8040 PRINT"probability of exactly  K  failures prior to  the  Rth  success"
8050 PRINT"when the probability of success on a single trial is  P
8060 PRINT : INPUT"Enter K";XK : K=XK
8070 IF K<0 OR K<>XK THEN PRINT"** ERROR **   K  MUST BE A POSITIVE INTEGER":GOTO 8060
8080 INPUT"Enter R";RA
8090 IF RA<0 OR RA<>INT(RA) THEN PRINT"** ERROR **   R  MUST BE A POSITIVE INTEGER":GOTO 8080
8100 INPUT"Enter P";PB
8110 IF PB=0 THEN PRINT"THE PROBABILITY OF";K;"FAILURES PRIOR TO";RA;"SUCCESSES = 1.0":GOTO 5170
8120 IF PB=1 THEN PRINT"THE PROBABILITY OF";K;"FAILURES PRIOR TO";RA;"SUCCESSES = 0.0":GOTO 5170
8130 IF PB<0 OR PB>1 THEN PRINT"** ERROR **   ";PB;"IS NOT A VALID PROBABILITY":GOTO 8100
8140 IF K=0 AND RA=0 THEN PRINT : PRINT"*** Reenter parameters":GOTO 8060
8150 IF K=0 THEN PR = PB[RA : GOTO 8200
8160 IF RA=0 THEN PR = (1-PB)[K : GOTO 8200
```

```
8170 TB=RA+K-1 : BB=K
8180 GOSUB 7430
8190 PR=BC * PB[RA * (1-PB)[K
8200 PRINT : PRINT"Prob. of exactly";K;"failures prior to";RA;"successes"
8210 PRINT"is equal to ";:PRINT USING"#.####";PR
8220 PRINT @960,"Do you want Cumulative Probability ";: KP=992 : L$="y" : R$="n" : GOSUB 600
8230 IF ZQ="N" OR ZQ="n" THEN PRINT @960, BL$; : GOTO 5170
8240 PR=0 : PRINT : INPUT"Enter K";XK
8250 K=XK : IF K<0 OR K<>XK THEN PRINT"** ERROR **  K MUST BE A POSITIVE INTEGER":GOTO 8240
8260 IF K=0 AND RA=0 THEN PRINT : PRINT"*** Reenter parameter":GOTO 8240
8270 IF K=0 THEN PR=PB[RA : GOTO 8330
8280 IF RA=0 THEN FOR I=1 TO K : PR=PR+(1-PB)[I : NEXT : GOTO 8330
8290 FOR BB=0 TO K
8300  TB=RA + BB - 1 : GOSUB 7430
8310 PR=PR + (BC * PB[RA * (1-PB)[BB)
8320 NEXT BB
8330 IF RA=1 ZX="st" : GOTO 8360
8340 IF RA=2 ZX="nd" : GOTO 8360
8350 IF RA=3 ZX="rd" ELSE ZX="th"
8360 PRINT : PRINT "Prob. of";K;"or fewer failures prior to the";RA; ZX;" success"
8370 PRINT"is equal to ";:PRINT USING"#.####";PR
8380 PRINT: GOTO 8220
```

POISSON DISTRIBUTION

```
9000 REM *** P O I S S O N   D I S T R I B U T I O N   S U B R O U T I N E
9010 IF I2=1 GOTO 9090
9020 CLS : PRINT TAB(20)"POISSON DISTRIBUTION
9030 PRINT : INPUT"Enter Rate (lambda)";LAM
9040 IF LAM<=0 PRINT"** ERROR **   ENTER A VALUE > 0":GOTO 9030
9050 INPUT"Enter time period (t)";TI
9060 IF TI<=0 PRINT"** ERROR **   ENTER A VALUE > 0":GOTO 9050
9070 INPUT"Enter number of occurrences";NC
9080 IF NC<0 OR NC<>INT(NC) PRINT"** ERROR **   ENTER A POSITIVE INTEGER VALUE":GOTO 9070
9090 PD=0
9100 IF NC<=1 GOTO 9140
9110 FOR PI=2 TO NC
9120 PD=PD + LOG(PI)
9130 NEXT PI
9140 PR=(-LAM*TI) + (NC*LOG(LAM*TI)) - PD
9150 PR=EXP(PR)
9160 IF I2=1 RETURN
9170 PRINT : PRINT"Prob. of exactly";NC;"occurrences in";TI;"time units = ";:PRINT USING"#.####";PR
9180 PRINT @960,"Do you want Cumulative Probability? ";: KP=992 : L$="y" : R$="n" : GOSUB 600
9190 IF ZQ="N" OR ZQ="n" THEN PRINT @960, BL$;: GOTO 5170
9200 PRINT : INPUT"Enter number of occurrences";NC
9210 IF NC<0 OR NC<>INT(NC) THEN PRINT"** ERROR **   ENTER A POSITIVE INTEGER VALUE":GOTO 9200
9220 IF NC=0 THEN PR=EXP(-LAM*TI):GOTO 9330
9230 PR=0
9240 FOR I=1 TO NC
9250 PU=0 : PD=0
9260 FOR PI=1 TO I
9270 PD=PD + LOG(PI)
9280 NEXT PI
9290 PU=(-LAM*TI) + (I*LOG(LAM*TI)) - PD
9300 PR=PR + EXP(PU)
9310 NEXT I
9320 PR=PR + EXP(-LAM*TI)
9330 IF I2=1 THEN RETURN
9340 PRINT : PRINT"Prob. of";NC;"or fewer occurrences"
9350 PRINT"in";TI;"time units = ";:PRINT USING "#.####";PR
9360 PRINT : GOTO 9180
```

EXPONENTIAL DISTRIBUTION

```
10000 REM *** E X P O N E N T I A L  D I S T R I B U T I O N  S U B R O U T I N E
10005 CLS : PRINT TAB(20)"EXPONENTIAL DISTRIBUTION"
10010 PRINT : PRINT"     An exponential distribution is often used in a practical" : PRINT "problem to represent the distribution of the time that elapses"
10015 PRINT"before the occurrence of some event.  Perhaps the most common"
10020 PRINT"example is the 'interarrival times ' between renewals in a"
10030 PRINT"Poisson Process.  A particularly nice feature of the exponen-"
10040 PRINT"tial distribution is its memoryless property.  That is, it is"
10050 PRINT"not necessary to consider past occurrences of an event in order"
10060 PRINT"to calculate probabilities for future occurrences of the event."
10070 PRINT : PRINT "     Press the SPACE BAR to proceed.";
10080 IF INKEY$ <> " " GOTO 10080
10090 CLS
10100 PRINT"NOTE: While the parameters and answers are expressed in terms"
10110 PRINT"of time, be aware that this is not the only context possible."
10120 PRINT : PRINT"You may use either the Exponential Rate, LAMBDA (1/MEAN) or the MEAN"
10130 KP=384 : L$="l" : R$="m" : GOSUB 600
10140 ZEP=ZQ
10150 PRINT : PRINT : PRINT"Enter ";:IF ZE="l" OR ZE="L" INPUT"LAMBDA";LAM ELSE INPUT"MEAN";LAM
10160 IF LAM<=0 THEN PRINT"** ERROR **    YOU MUST ENTER A VALUE > 0":GOTO 10150
10170 IF ZE="M" OR ZE="m" THEN LAM=1/LAM
10180 INPUT"Enter number of time units";TI
10190 PRINT : PRINT"Prob. of";TI;"time units or less until the next" : PRINT"occurrence is =";:PRINT USING"#.####";1-EXP(-LAMBDA*TI)
10200 PRINT : PRINT"Prob. of more than";TI;"time units until the next" : PRINT"occurrence is =";:PRINT USING"#.####";EXP(-LAMBDA*TI)
10210 PRINT : PRINT"Expected time until the next occurrence is";1/LAMBDA;:PRINT" time units"
10220 PRINT @960, BL$ : GOTO 5170
```

CHI - SQUARE DISTRIBUTION

```
11000 REM *** C H I - S Q U A R E  D I S T R I B U T I O N  S U B R O U T I N E
11010 IF I2=1 GOTO 11080
11020 CLS: PRINT @20,"CHI-SQUARE DISTRIBUTION"
11030 PRINT @192,"";:INPUT"Enter Chi-Square Statistic  x ";CX
11040 IF CX<=0 PRINT"** ERROR **   CHI-SQUARE STATISTIC MUST BE > 0":GOTO 11030
11050 IF I2=0 THEN PX=CX
11060 INPUT"Enter Degrees of Freedom";DF
11070 IF DF<=0 OR DF<>INT(DF) THEN PRINT"** ERROR **   ENTER AN INTEGER > 0":GOTO 11060
11080 IF DF>30 GOTO 11290
11090 DP=1
11100 FOR I=INT(DF) TO 2 STEP -2
11110 DP=DP*I
11120 NEXT I
11130 NP=CX[(INT((DF+1)/2))*EXP(-CX/2)/DP
11140 IF INT(DF/2)=DF/2 GOTO 11170
11150 PC=SQR(2/CX/3.141593)
11160 GOTO 11180
11170 PC=1
11180 LC=1
11190 MC=1
11200 D=DF
11210 D=D+2
11220 MC=MC*CX/D
11230 IF MC<.000001 GOTO 11260
11240 LC=LC+MC
11250 GOTO 11210
11260 FOX=PC*NP*LC
11270 QOX=1-FOX
11280 RETURN
```

```
11290 NX=((CX/DF)[(1/3) - (1-(2/(9*DF))))/SQR(2/(9*DF))
11300 MU=0 : VAR=1 : I2=1
11310 GOSUB 6000
11320 RETURN
```

STUDENT's - t DISTRIBUTION

```
12000 REM  *** t-DISTRIBUTION SUBROUTINE
12010 IF I2=1 GOTO 12060
12020 CLS:PRINT @20,"T-DISTRIBUTION"
12030 PRINT : PRINT : INPUT"Enter t-Statistic x ";TX
12040 INPUT"Enter Degrees of Freedom";DF
12050 IF DF<=0 OR DF<>INT(DF) THEN PRINT"** ERROR **  ENTER AN INTEGER > 0":GOTO 12040
12060 IF TX<0 I3=1 ELSE I3=0
12070 IF DF>25 GOTO 12500
12080 THETA=ATN(ABS(TX)/SQR(DF))
12090 IF DF<>1 GOTO 12140
12100 AT=.6366198*THETA
12110 FOX=(AT+1)/2
12120 QOX=1-FOX
12130 GOTO 12450
12140 IF INT(DF/2)=DF/2 GOTO 12270
12150 IF DF=3 GOTO 12420
12160 TS=0 : TP=1 : TC=COS(THETA)
12170 FOR I=1 TO INT((DF-3)/2)
12180 TP=TP*2*I/(1+2*I)
12190 TC=TC*(COS(THETA)[2)
12200 TS=TS+(TP*TC)
12210 NEXT I
12220 TS=TS+COS(THETA)
12230 AT=.6366198*(THETA+(SIN(THETA)*TS))
12240 FOX=(AT+1)/2
12250 QOX=1-FOX
12260 GOTO 12450
12270 IF DF<>2 GOTO 12320
12280 AT=SIN(THETA)
12290 FOX=(AT+1)/2
12300 QOX=1-FOX
12310 GOTO 12450
12320 TS=0 : TP=1 : TC=1
12330 FOR I=1 TO INT((DF-2)/2)
12340 TP=TP*((2*I)-1)/(2*I)
12350 TC=TC*(COS(THETA)[2)
12360 TS=TS+(TP*TC)
12370 NEXT I
12380 AT=SIN(THETA)*(1+TS)
12390 FOX=(AT+1)/2
12400 QOX=1-FOX
12410 GOTO 12450
12420 AT=.6366198*(THETA+(SIN(THETA)*COS(THETA)))
12430 FOX=(AT+1)/2
12440 QOX=1-FOX
12450 PX=TX
12460 IF I3=0 RETURN
12470 QOX=FOX
12480 FOX=1-QOX
12490 RETURN
12500 NX=( TX * (1 - .25/DF) )/SQR(1 + .5*TX[2/DF)
12510 MU=0 : VAR=1 : I2=1 : GOSUB 6000
12520 PX=TX
12530 AT=2*FOX - 1
12540 RETURN
```

F - DISTRIBUTION

```
13000 REM *** F-DISTRIBUTION SUBROUTINE
13010 IF I2=1 GOTO 13110
13020 CLS : PRINT @20,"F-DISTRIBUTION"
13030 PRINT @128,"";: INPUT"Enter F-Statistic x ";FX
13040 IF FX<0 THEN PRINT@128, CHR$(253);:PRINT @128,"F-STATISTIC MUST BE A POSITIVE NUMBER";:FOR I
=1 TO 1500:NEXT:PRINT @128,CHR$(253);:GOTO 13030
13050 INPUT"Enter Degrees of Freedom in Numerator";DN
13060 IF DN>0 AND DN=INT(DN) GOTO 13080
13070 PRINT"** ERROR **   ENTER AN INTEGER > 0":GOTO 13050
13080 INPUT"Enter Degrees of Freedom in Denominator";DD
13090 IF DD>0 AND DD=INT(DD) GOTO 13110
13100 PRINT"** ERROR **  ENTER AN INTEGER > 0":GOTO 13080
13110 PX=FX : IF FX=0 THEN FOX=0 : QOX=1 : RETURN
13120 IF DN > 60 AND DD > 60 GOTO 13720
13130 REM *** APPROX. FOR SMALL TO MODERATE D.F. ***
13140 K1=DN : K2=DD : PX=FX
13150 A=DN*FX/(DN*FX+DD) : A1=1-A
13160 IF A1< 1E-37 THEN A1=1E-37
13170 D1=DN*.5 : D2=DD*.5 : D3=D1+D2-1
13180 R=0 : S1=0 : S2=0 : DL=1 : C=.25
13190 XM=1 : XX=1 : PI=3.141593 : KN=DD
13200 KM=INT(D2)*2
13210 IF KM<>KN GOTO 13390
13220 KN=D2-1
13230 IF KN=0 GOTO 13320
13240 FOR I=1 TO KN
13250 S1=DL+S1*R
13260 D2=D2-1
13270 D3=D3-1
13280 TEM=A1/D2
13290 R=D3*TEM
13300 S2=(R+TEM)*S2
13310 NEXT I
13320 S1=DL+S1*R
13330 DL=0
13340 T=-1
13350 D3=-1
13360 S2=A*S2
13370 C=C+.5
13380 GOTO 13570
13390 KN=INT(D2)
13400 IF KN=0 GOTO 13490
13410 FOR I=1 TO KN
13420 S1=DL+S1*R
13430 D2=D2-1
13440 D3=D3-1
13450 TEM=A1/D2
13460 R=D3*TEM
13470 S2=(R+TEM)*S2
13480 NEXT I
13490 S1=XX*S1
13500 S2=XX*S2
13510 ART=SQR(A1)
13520 XM=XM*ART
13530 T=(XM-ART)/A1
13540 D3=-.5
13550 XX=2/PI
13560 C=C*2
13570 IF C>.875 GOTO 13660
13580 D2=D1
13590 D3=D2+D3
13600 S2=S1
```

```
13610 S1=0
13620 A1=A
13630 IF A1< 1E-37 THEN A1=1E-37
13640 KN=DN
13650 GOTO 13200
13660 IF C<1.125 THEN DL=4/PI * ATN(T)
13670 FOX=XM*(S2-S1)-DL
13680 IF FOX<0 THEN FOX=0
13690 IF FOX>1 THEN FOX=1
13700 QOX=1-FOX
13710 RETURN
13720 REM *** A P P R O X.  F O R  L A R G E  D. F.  ***
13730 NX=(((FX[(1/3))*(1-(2/(9*DD))))-(1-(2/(9*DN))))/SQR((2/(9*DN))+(FX[(2/3))*2/(9*DD))
13740 MU=0 : VAR=1 : I2=1
13750 GOSUB 6000
13760 PX=FX
13770 RETURN
```

INVERSE NORMAL

```
14000 REM  ***  I N V E R S E   N O R M A L   S U B R O U T I N E
14010 IF I2=1 GOTO 14100
14020 CLS : PRINT @20, "INVERSE NORMAL"
14030 PRINT : PRINT : INPUT"Enter Mean";MO
14040 INPUT "Enter Variance";VR
14050 IF VR<=0 PRINT"** ERROR **  ";VR;"IS NOT A VALID VARIANCE":GOTO 14040
14060 INPUT"Enter Probability (i.e.  F(x) =  Prob. ( X <= x ) = ";PR
14070 IF PR=0 THEN XQ=9E-36 ELSE IF PR=1 XQ=9E+36
14080 IF PR=0 OR PR=1 RETURN
14090 IF PR<0 OR PR>1 THEN PRINT"** ERROR **  ";PR;"IS NOT A VALID PROBABILITY":GOTO 14060
14100 IF PR>.5 THEN RL=SQR(LOG(1/((1-PR)[2)))  ELSE RL=SQR(LOG( 1/(PR[2)))
14110 XQ=RL-((2.515517+(.802853*RL)+(.010328*RL[2))/(1+(1.432788*RL)+(.189269*RL[2)+(.001308*RL[3)))
14120 IF PR>=.5 THEN XQ=MO+(SQR(VR)*XQ) ELSE XQ=MO-(SQR(VR)*XQ)
14130 RETURN
```

INVERSE CHI - SQUARE DISTRIBUTION

```
15000 REM  ***  I N V E R S E   C H I - S Q U A R E   S U B R O U T I N E
15010 IF I2=1 GOTO 15090
15020 CLS : PRINT @20,"INVERSE CHI-SQUARE"
15030 PRINT : INPUT"Enter Degrees of Freedom";DF
15040 IF DF<=0 OR DF<>INT(DF) THEN PRINT"** ERROR **  ENTER AN INTEGER > 0":GOTO 15030
15050 INPUT "Enter Probability  i.e. F(x) = Prob. ( X <= x ) = ";PR
15060 IF PR=0 THEN XQ=0 : RETURN
15070 IF PR=1 THEN XQ=9E36 : RETURN
15080 IF PR<0 OR PR>1 THEN PRINT"** ERROR **  ";PR;"IS NOT A VALID PROBABILITY":GOTO 15050
15090 MO=0 : VR=1 : I2=1 : PZ=PR
15100 IF DF>1 GOTO 15150
15110 PR=.5*(1-PZ)
15120 GOSUB 14000
15130 XQ=XQ[2 : PR=PZ
15140 RETURN
15150 IF DF>2 GOTO 15180
15160 XQ=-2*LOG(1-PZ)
15170 RETURN
15180 GOSUB 14000
15190 DQ=XQ
15200 IF DF>(2+INT(4*ABS(DQ))) GOTO 15300
15210 X9=1.0000886 - .2237368/DF - .01513904/DF[2
15220 X9=X9+(DF[-.5 * DQ * (.4713941 + .02607083/DF - .008986007/DF[2))
15230 X9=X9 + (1/DF * DQ[2 * (.0001348028 + .01128186/DF + .02277679/DF[2))
15240 X9=X9 + (DF[-1.5 * DQ[3 * (-.008553069 - .01153761/DF - .01323293/DF[2))
```

```
15250 X9=X9 + (DF[-2 * DQ[4 * (.00312558 + .005169654/DF - .006950356/DF[2))
15260 X9=X9 + (DF[-2.5 * DQ[5 * (-.0008426812 + .00253001/DF + .001060438/DF[2))
15270 X9=X9 + (DF[-3 * DQ[6 * (.00009780499 - .001450117/DF + .001565326/DF[2))
15280 XQ=DF*X9[3
15290 RETURN
15300 X9=1 - 2/(9*DF) + (4*DQ[4 + 16*DQ[2 -28)/(1215*DF[2)
15310 X9=X9 + (8*DQ[6 + 720*DQ[4 + 3126*DQ[2 + 2904)/(229635*DF[illegible])
15320 X8=DQ/3 + (-DQ[3 + 3*DQ)/(162*DF)
15330 X8=X8 - (3*DQ[5 + 40*DQ[3 + 45*DQ)/(5832*DF[2)
15340 X8=X8 + (301*DQ[7 - 1517*DQ[5 -32769*DQ[3 - 79349*DQ)/(7873200*DF[3)
15350 X8= SQR(2/DF) * X8
15360 XQ=DF * ((X8+X9)[3)
15370 RETURN
```

<u>**INVERSE - t**</u>

```
16000 REM *** I N V E R S E - t  S U B R O U T I N E
16010 IF I2=1 GOTO 16090
16020 CLS : PRINT @16, "INVERSE-t PROBABILITY DISTRIBUTION"
16030 PRINT : INPUT"Enter Degrees of Freedom";DF
16040 IF DF<=0 OR DF<>INT(DF) THEN PRINT"** ERROR **   ENTER AN INTEGER > 0":GOTO 16030
16050 INPUT"Enter Probability  i.e. F(x) = Prob. ( X <= x ) = ";PR
16060 IF PR=1 THEN XQ=9E36 ELSE IF PR=0 XQ=9E-36
16070 IF PR=1 OR PR=0 RETURN
16080 IF PR<0 OR PR>1 THEN PRINT"** ERROR **   ";PR;"IS NOT A VALID PROBABILITY":GOTO 16050
16090 MO=0 : VR=1 : I2=1
16100 GOSUB 14000
16110 XC=XQ
16120 XQ=(.25 * (XC[3 + XC))/DF
16130 XQ=XQ + ((1/96) * (5*XC[5 + 16*XC[3 + 3*XC))/DF[2
16140 XQ=XQ + ((1/384) * (3*XC[7 + 19*XC[5 + 17*XC[3 -15*XC))/DF[3
16150 XQ=XQ + ((1/92160) * (79*XC[9 + 776*XC[7 + 1482*XC[5 -1920*XC[3 -945*XC))/DF[4
16160 XQ= XC + XQ
16170 IF DF>3 RETURN
16180 IF PR>.05 AND PR<.95 GOTO 16230
16190 IF DF>1 GOTO 16210ELSE IF PR>.999 OR PR<.001 THEN DELTA=1000 ELSE DELTA=100
16200 GOTO 16240
16210 IF DF=2 AND PR>.99 THEN DELTA=100 ELSE DELTA =10
16220 GOTO 16240
16230 DELTA=1
16240 IF PR>=.5 THEN TX=XQ ELSE TX=-XQTX=XQ
16250 IF PR>=.5 THEN PE=PR ELSE PE=1-PR
16260 GOSUB 12000
16270 IF ABS(PE-FOX) >= .00001 THEN GOTO 16290ELSE IF PR=PE THEN XQ=TX ELSE XQ=-TX
16280 RETURN
16290 IF PE < FOX THEN TX=TX-DELTA
16300 IF PE < FOX THEN DELTA=DELTA/10
16310 IF PE < FOX THEN GOTO 16260
16320 TX=TX+DELTA
16330 GOTO 16260
```

INVERSE F

```
17000 REM *** I N V E R S E - F  S U B R O U T I N E
17010 IF I2=1 GOTO 17150
17020 CLS : PRINT @ 22, "I N V E R S E - F"
17030 PRINT : PRINT
17040 INPUT"Enter Degrees of Freedom in Numerator";DN
17050 IF DN>0 AND DN=INT(DN) GOTO 17070
17060 PRINT"** ERROR **   ENTER AN INTEGER > 0":GOTO 17040
17070 INPUT"Enter Degrees of Freedom in Denominator";DD
17080 IF DD>0 AND DD=INT(DD) GOTO 17100
17090 PRINT"** ERROR **   ENTER AN INTEGER > 0":GOTO 17070
17100 INPUT"Enter Probability  i.e. F(x) = Prob. ( X <= x ) = ";PR
17110 IF PR<0 OR PR>1 THEN PRINT"** ERROR **  ";PR;"IS NOT A VALID PROBABILITY":GOTO 17100
17120 IF PR>.999 PRINT "THIS SUBROUTINE DOES NOT CALCULATE QUANTILES CORRESPONDING TO  PROBA
BILITIES GREATER THAN .999":GOTO 17100
17130 IF PR=1 THEN PR=.9999
17140 IF PR=0 THEN XQ=0 : RETURN
17150 IF DN<20 OR DD<20 GOTO 17230
17160 MO=0 : VR=1 : I2=1
17170 GOSUB 14000
17180 H = 2 * ( (1/(DD-1) + 1/(DN-1))[-1 )
17190 LM = (XQ[2 - 3)/6
17200 WX=(XQ * SQR(H + LM))/H - ( (1/(DN-1) - 1/(DD-1) ) * ( LM + 5/6 - 2/(3*H) ))
17210 XQ = EXP(2*WX)
17220 RETURN
17230 IF DN<>1 AND DD<>1 OR DN=1 AND DD=1  GOTO 17330
17240 PH=PR : IF DN=1 THEN PR=(1+PR)/2 ELSE PR=1-PR/2
17250 IF DN=1 THEN DF=DD ELSE DF=DN
17260 I2=1 : GOSUB 16000
17270 XQ=XQ*XQ : IF DD=1 THEN XQ=1/XQ
17280 PR=PH
17290 IF PR>=.99 THEN XQ=.994*XQ : GOTO 17320
17300 IF PR>=.975 AND DN>3 THEN XQ=.995*XQ : GOTO 17320
17310 IF PR>=.95 AND DN>3 THEN XQ=.996*XQ
17320 RETURN
17330 IF PR>.999 AND DD=1 THEN XQ=9E36 : RETURN
17340 IF PR>=.997 AND DD=1THEN DELTA=500000 : GOTO 17410
17350 IF PR>=.995 AND DD=1 THEN DELTA=15000 : GOTO 17410
17360 IF PR>=.99 AND DD=1 THEN DELTA=5000 : GOTO 17410
17370 IF PR>=.975 AND DD=1 OR PR>=.999 AND DD=2 THEN DELTA=1000 : GOTO 17410
17380 IF PR>=.9 AND DD=1 OR PR>.99 AND DD<=2 OR PR>=.999 AND DD<=3 THEN DELTA=100 : GOTO 17410
17390 IF PR>=.99 AND DD<=3 THEN DELTA=50 : GOTO 17410
17400 IF PR>=.975 AND DD<=2 THEN DELTA=50 ELSE DELTA=10
17410 LL=0
17420 FX=DELTA
17430 I2=1 : GOSUB 13000
17440 IF ABS(PR-FOX)<=.00001 THEN XQ=FX : RETURN
17450 IF PR < FOX THEN DELTA=DELTA/2 : FX=DELTA : GOTO 17430
17460 LL=FX : FX=FX+DELTA : UL=FX : GOSUB 13000
17470 IF ABS(PR-FOX)<=.00001 THEN XQ=FX : RETURN
17480 DELTA=(UL-LL)/2
17490 FX=LL+DELTA
17500 GOSUB 13000
17510 IF ABS(PR-FOX)<=.00001 THEN XQ=FX : RETURN
17520 IF PR > FOX THEN LL=FX ELSE UL=FX
17530 GOTO 17480
```

INVERSE EXPONENTIAL

```
18000 REM *** I N V E R S E  E X P O N E N T I A L  S U B R O U T I N E
18010 CLS : PRINT @20,"INVERSE EXPONENTIAL"
18020 PRINT : PRINT"    You may either enter the exponential parameter (LAMBDA) or"
18030 PRINT"the exponential MEAN (1/lambda)."
18040 KP=320 : L$="l" : R$="m" : GOSUB 600
18050 ZEP=ZQ
18060 IF ZEP="l" OR ZEP="L" THEN ZEP="LAMBDA" ELSE ZEP="MEAN"
18070 PRINT : PRINT"Enter ";ZEP;
18080 INPUT LAMBDA
18090 IF LAM <= 0 THEN PRINT"** ERROR **   ENTER A VALUE > 0":GOTO 18070
18100 PRINT : INPUT"Enter Probability  i.e. Prob.( T <= t ) =";PR
18110 IF PR=0 THEN TI=0 : PR=1 : GOTO 18160
18120 IF PR=1 THEN PRINT : PRINT"For Prob. ( T <= t ) = 1.0     t = infinity":RETURN
18130 IF PR<0 OR PR>1 THEN PRINT"** ERROR **   ";PR;"IS NOT A VALID PROBABILITY":GOTO 18100
18140 PB=1-PR
18150 IF ZEP="LAMBDA" TI= - LOG(PB)/LAMBDA ELSE TI = - LOG(PB)*LAMBDA
18160 PRINT : PRINT"For Prob. ( T <= t ) =";PR;"    t =";TI
18170 RETURN
```

END OF PROGRAM #1

PROGRAM #2

MENU

```
10 REM  *** M A I N  M E N U ***
20 CLS : PRINT TAB(20) "***  MENU  ***"
30 PRINT
40 PRINT"(1) Sample Statistics (mu, sigma, g1, g2)
50 PRINT"(2) Seven Number Summary
60 PRINT"(3) Histogram, Stem & Leaf, Probability Plots
70 PRINT"(4) Return to Program Selection Menu
80 PRINT @645,"NUMBER OF OPTION DESIRED ";CHR$(123);CHR$(176);CHR$(125)
90 ZI=INKEY$ : IF ZI="" GOTO 90ELSE IO=VAL(ZI)
100 PRINT @671,ZI;: FOR I=1 TO 100: NEXT
110 IF IO<1 OR IO>4 THEN PRINT @640,CHR$(253) : PRINT @640,"***  NOT A VALID OPTION  ***": FOR I=1 TO
1000 : NEXT : PRINT @640, CHR$(253) : GOTO 80
120 ON IO GOSUB 500, 1000, 2000, 10000
130 GOTO 10
140 CLS
150 PRINT"Option";IO;"is not applicable since only sample statistics were   entered."
160 PRINT@960,"Press SPACE BAR to return to main menu";
170 PRINT "Kurtosis =";G2
180 IF INKEY$<>" " GOTO 180
190 CLS
200 GOTO 10
```

DISPLAY SAMPLE STATISTICS

```
500 REM  * * * S A M P L E   S T A T I S T I C S   S U B R O U T I N E
510 CLS
520 PRINT TAB(23)"SAMPLE STATISTICS"
530 PRINT @192,"Sample Mean =";XB
540 PRINT "Sample Variance (unbiased estimate) =";V1
550 PRINT"M.L.E. for Variance =";V
560 PRINT "Std. Deviation (sqr. root of sample var.) =";SQR(V1)
570 PRINT "Skewness =";G1
580 PRINT "Kurtosis =";G2
590 PRINT : PRINT"NOTE:  The unbiased estimate of variance "
600 PRINT"      ";CHR$(123);" i.e. SUM (X(i) - Xbar)[2 / (n-1) ";CHR$(125);" is used in"
610 PRINT"      all subsequent calculations throughout the package."
620 PRINT @960,"PRESS SPACE BAR TO RETURN TO MAIN MENU";
630 IF INKEY$ <> " " GOTO 630
640 RETURN
```

SEVEN NUMBER SUMMARY

```
1000 REM  * * * S E V E N   N U M B E R   S U M M A R Y   S U B R O U T I N E
1010 CLS
1020 PRINT TAB(11)"S E V E N   N U M B E R   S U M M A R Y"
1030 MD=INT(.5*(IN+1)) : QD=INT(.5*(MD+1)) : ED=INT(.5*(QD+1))
1040 IF IN/2=INT(IN/2) THEN M=.5*(Y(MD)+Y(MD+1)) ELSE M=Y(MD)
1050 IF QD=.5*(MD+1) THEN QL=Y(QD) : QU=Y(IN-QD+1) : GOTO 1070
1060 QL=.5*(Y(QD)+Y(QD+1)) : QU=.5*(Y(IN-QD)+Y(IN-QD+1))
1070 IF ED=.5*(QD+1) THEN EL=Y(ED) : EU=Y(IN-ED+1) : GOTO 1090
1080 EL=.5*(Y(ED)+Y(ED+1)) : EU=.5*(Y(IN-ED)+Y(IN-ED+1))
1090 TV=M : GOSUB 1250 : PRINT @146,"MEDIAN = ";: PRINT USING P$;M;
1100 PRINT @237,"MEASURES OF";
1110 PRINT @267,"LOWER    MEAN    UPPER    SKEWNESS"
1115 PRINT @331,"-----    ----    -----    --------"
1120 PRINT @384,"QUARTER:" : TV=QL : GOSUB 1250 : PRINT @394,"";:PRINT USING P$;QL;
1130 TV=.5*(QL+QU) : GOSUB 1250 : PRINT @405,"";:PRINT USING P$; TV;
```

```
1140 TV=QU : GOSUB 1250 : PRINT @416,"";:PRINT USING P$;QU;
1150 TV= (.5*(QU+QL) - M)/(QU-QL) : GOSUB 1250 : PRINT @431,"";:PRINT USING P$;TV;
1155 PRINT @512,"EIGTH:";: TV=EL : GOSUB 1250 : PRINT @522,"";:PRINT USING P$;EL;
1160 TV=.5*(EL+EU) : GOSUB 1250 : PRINT @533,"";:PRINT USING P$;TV;
1165 TV=EU : GOSUB 1250 : PRINT @544,"";:PRINT USING P$;EU;
1170 TV=(.5*(EL+EU) - M)/(EU-EL) : GOSUB 1250 : PRINT @559,"";:PRINT USING P$;TV;
1175 PRINT @640,"EXTREME:"; : TV=Y(1) : GOSUB 1250 : PRINT @650,"";:PRINT USING P$; Y(1);
1180 TV=.5*(Y(1)+Y(IN)) : GOSUB 1250 : PRINT @661,"";:PRINT USING P$;TV;
1190 TV=Y(IN) : GOSUB 1250 : PRINT @672,"";:PRINT USING P$;Y(IN);
1200 TV=(.5*(Y(IN)+Y(1)) - M)/(Y(IN)-Y(1)) : GOSUB 1250 : PRINT @687,"";:PRINT USING P$;TV
1210 PRINT @896,"Press SPACE BAR when ready to return to program menu";
1220 IF INKEY$<>" " THEN 1220
1230 RETURN
1250 HB=ABS(TV)
1255 P1$(1)="#.####" : P1$(2)="##.###" : P1$(3)="###.##" : P1$(4)="####.#" : P1$(5)="#####" : P1$(6)="####
[[[["
1260 IF HB>=.0001 AND HB<1  OR HB=0 THEN P$=P1$(1) : RETURN
1270 IF HB>=1 AND HB<10 THEN P$=P1$(2) : RETURN
1280 IF HB>=10 AND HB<100 THEN P$=P1$(3) : RETURN
1290 IF HB>=100 AND HB<1000 THEN P$=P1$(4) : RETURN
1300 IF HB>=1000 AND HB<10000 THEN P$=P1$(5) : RETURN
1310 P$=P1$(6) : RETURN
```

<u>**MENU OF DATA PLOTS**</u>

```
2000 REM  *** D A T A  P L O T S  S U B R O U T I N E
2010 CLS
2020 PRINT TAB(23)"D A T A   P L O T S"
2030 PRINT : PRINT
2040 PRINT "MENU:" : PRINT"   (1) Histogram
2050 PRINT"   (2) Stem & Leaf Plot
2060 OC=CI
2070 PRINT"   (3) Normal Probability Plot
2080 PRINT"   (4) Exponential Probability Plot
2090 PRINT @645,"NUMBER OF OPTION DESIRED ";CHR$(123);CHR$(176);CHR$(125)
2100 ZI=INKEY$ : IF ZI="" GOTO 2100ELSE IO=VAL(ZI)
2110 PRINT @671,ZI;: FOR I=1 TO 100 : NEXT
2120 IF IO<1 OR IO>4 PRINT @640, CHR$(250);: PRINT @640,"*** NOT A VALID OPTION ***";:FOR I=1 TO 1000:
NEXT : PRINT @640,CHR$(253) : GOTO 2090
2130 IF IO>=3 GOTO 2200
2140 PRINT @640,"Min Value =";Y(1);"     Max Value =";Y(IN):PRINT"            Range =";Y(IN)-Y(1)
2150 PRINT : PRINT"AFTER VIEWING ";: IF IO=1 PRINT"HISTOGRAM"; ELSE PRINT"STEM & LEAF";
2160 PRINT" PRESS SPACE BAR TO CONTINUE";
2170 IF IN<=50 OR IO=2 THEN FOR I=1 TO 2000 : NEXT
2180 SC=10[(INT(LOG(Y(IN)-Y(1))/LOG(10)))
2190 NB=0 : CI=1
2200 ON IO GOSUB 3000, 4000, 5000, 6000
2210 RETURN
```

<u>**HISTOGRAM**</u>

```
3000 REM  *** H I S T O G R A M  S U B R O U T I N E
3010 NB=0
3020 IF SC<1 THEN GF=INT(Y(1)/SC) ELSE GF=INT(Y(1)/INT(SC))
3030 IF SC<1 GOTO 3060
3040 IF GF + NB*CI < INT(Y(IN)/INT(SC)) + 1E-6 AND NB <= 15 THEN NB=NB+1 : GOTO 3040
3050 GOTO 3070
3060 IF GF + NB*CI < (INT(Y(IN)/SC) + 1E-6) AND NB <= 15 THEN NB=NB+1 : GOTO 3060
3070 PC=NB/15 : IF PC>1 THEN CI=CI+1 : NB=0 : GOTO 3020
3080 IF BF$<>"ON" AND PC<1/3 THEN CI=CI/2 : NB=0 : GOTO 3020
3090 II=1
3100 FOR I=1 TO NB : NI(I)=0
3110 FOR J=II TO IN : IF Y(J)/SC < I*CI + INT(Y(1)/SC) - 1E-6 THEN NI(I)=NI(I)+1 ELSE II=J : GOTO 3130
3120 NEXT J
```

```
3130 IF II > IN GOTO 3150
3140 NEXT I
3150 MN=0 : FOR I=1 TO NB : IF NI(I) > MN THEN MN=NI(I)
3160 NEXT I
3170 IF MN < 11 THEN VS=.25 : GOTO 3200
3180 IF MN < 22 THEN VS=.5 : GOTO 3200
3190 IF MN <= 45 THEN VS=1 ELSE VS=INT(MN/45)+1
3200 IF SC < 1 GOTO 3260
3210 CLS : CN=INT(Y(1)/INT(SC)) : FOR I=0 TO 4*(NB) STEP 4
3220 TN=CN-INT(CN) : IF TN=0 OR TN=.5 THEN PRINT @961+I, CN;
3230 CN=CN+CI
3240 IF CN/SC > 15*CI + INT(Y(1)/SC) GOTO 3310
3250 GOTO 3300
3260 CLS : CN=INT(Y(1)/SC) : FOR I = 0 TO 4*NB STEP 4
3270 TN=CN-INT(CN) : IF TN=0 OR TN=.5 THEN PRINT @961+I, CN;
3280 CN=CN+CI
3290 REM  IF CN > 15*CI + INT(Y(1)/SC) GOTO 1882
3300 NEXT I
3310 IF VS>=1 THEN LN=2 ELSE LN=0
3320 FOR I=0 TO 14
3330 IF VS>=1 THEN PRINT @896-I*64, MID$(STR$(VS*LN),2,3)+"-";: LN=LN+3 : GOTO 3370
3340 IF I=2 THEN K=2 ELSE IF I=6 THEN K=5 ELSE IF I=10 THEN K=8 ELSE IF I=14 THEN K=11
3350 IF VS=.5 THEN K=2*K
3360 IF I=2 OR I=6 OR I=10 OR I=14 THEN PRINT @896-I*64,MID$(STR$(K),2,2)+"-";
3370 NEXT I
3380 X1=9 : FOR I=1 TO NB : IF NI(I)=0 GOTO 3440
3390 JU=NI(I)/VS - 1 : EC=NI(I)/VS-JU : IF EC<.5 THEN GOTO 3410
3400 IF VS>2 THEN JU=JU+1
3410 FOR J=0 TO JU : Y1=44-J : IF Y1<0 THEN Y1=0
3420 SET(X1,Y1) : SET(X1-1,Y1) : SET(X1-2,Y1) : SET(X1+2,Y1) : SET(X1+1,Y1) : SET(X1-3,Y1) : IF X1<124 THEN S
ET(X1+3,Y1)
3430 NEXT J
3440 X1=X1+8 : NEXT I
3450 IF SC<>1 THEN PRINT@38,"Actual Values are";VAL(STR$(SC));: PRINT @102,"times horizontal values";
3460 IF INKEY$ <> " " GOTO 3460
3470 CLS : PRINT"OPTIONS:" : PRINT"   (1) Return to Menu" : PRINT"   (2) Redraw Histogram with larger class in
terval" : IF NB<=7 PRINT"   (3) Redraw with smaller class interval"
3480 PRINT @453,"NUMBER OF OPTION DESIRED ";CHR$(123);CHR$(176);CHR$(125)
3490 ZI=INKEY$ : IF ZI="" GOTO 3490ELSE IO=VAL(ZI)
3500 PRINT @479,ZI;: FOR I=1 TO 100 : NEXT
3510 IF NB>7 THEN IU=2 ELSE IU=3
3520 IF IO<1 OR IO>IU PRINT @448,CHR$(253) : PRINT @448,"***  NOT A VALID OPTION  ***": FOR I=1 TO 100
0 : NEXT : PRINT @448,CHR$(253) : GOTO 3480
3530 IF IO=2 THEN BF$="ON" ELSE BF$="OFF"
3540 IF IO=1 THEN RETURN ELSE IF IO=2 THEN CI=2*CI ELSE CI=CI/2
3550 CLS : PRINT"AFTER VIEWING HISTOGRAM, PRESS SPACE BAR TO CONTINUE";
3560 NB=0 : GOTO 3020
```

STEM & LEAF

```
4000 REM  ***  S T E M  &  L E A F   S U B R O U T I N E  ***
4010 CLS : IF SC<>1 THEN PRINT"ACTUAL VALUES ARE";SC;"TIMES THOSE SHOWN"
4020 SP=INT(Y(1)/SC+1E-6)
4030 IF SP<0 THEN SP=SP-CI
4040 EP=INT(Y(IN)/SC) + CI
4050 IF ABS(EP-SP)/CI<=3 THEN CI=CI/2 : GOTO 4010
4060 I6=0 : JC=1 : ZP=""
4070 FOR CO=SP TO EP STEP CI
4080 IF CO < 0 THEN CJ=INT(CO)+1  ELSE  CJ=INT(CO)
4090 IF CJ = 0  AND CO < 0 GOTO 4120
4100 PRINT USING"+##";CJ; : PRINT CHR$(124);
4110 GOTO 4130
4120 PRINT" -0";CHR$(124);
4130 FOR J=JC TO IN
4140 IF Y(1) <> INT( Y(1) ) GOTO 4170
```

```
4150 IF Y(1)/SC > CO+CI GOTO 4350
4160 GOTO 4180
4170 IF Y(1)/SC + 1E-6 >= CO +CI THEN GOTO 4350
4180 IF Y(J) >= 0 GOTO 4230
4190 IF Y(J) <> INT (Y(J)) GOTO 4230
4200 IF Y(J)/SC - .001 <= CO+1 AND Y(J)/SC > CO - CI + 1.001 THEN SL=(10*(Y(J)/SC - INT(Y(J)/SC+1E-6))) ELS
E GOTO 4310
4210 IF I6=0 THEN I6=1
4220 GOTO 4250
4230 IF Y(J)/SC+1E-6 >= CO AND Y(J)/SC+1E-6  < CO+CI THEN SL=(10*(Y(J)/SC -INT(Y(J)/SC+1E-6)) ) ELSE GOT
O 4310
4240 IF I6=0 THEN I6=1
4250 IF Y(J)<0  AND SL <> 0 THEN SL=10-SL
4260 ZS=MID$( STR$(SL) ,2,1)
4270 IF ZS="." THEN ZS="0"
4280 IF CO<0 THEN ZP=ZS + ZP ELSE ZP=ZP+ZS
4290 JC=JC+1 : IF JC > IN THEN JC=IN
4300 GOTO 4330
4310 IF I6=1 THEN I6=2
4320 IF I6=2 GOTO  4350
4330 NEXT J
4340 I6=0
4350 PRINT ZP : ZP="" : I6=0 :  NEXT CO
4360 PRINT@960,"PRESS SPACE BAR TO CONTINUE";
4370 IF INKEY$ <> " " GOTO 4370
4380 IF ABS((EP-SP)/CI)>8 THEN RETURN ELSE CLS: PRINT"Do you want to see the Stem & Leaf with a smaller
Class Interval"
4390 PRINT @197,CHR$(123);" Y or N ";CHR$(125);"  ";CHR$(123);CHR$(176);CHR$(125)
4400 ZI=INKEY$ : IF ZI="" GOTO 4400ELSE PRINT @210,ZI;: FOR I=1 TO 100 : NEXT
4410 IF ZI="y" OR ZI="Y" THEN CI=CI/2 : GOTO 4000
4420 IF ZI<>"n" AND ZI<>"N" THEN GOTO 4390
4430 RETURN
```

NORMAL PROBABILITY PLOT

```
5000 REM  ***  N O R M A L   P R O B A B I L I T Y   P L O T   S U B R O U T I N E
5010 CLS
5020 PRINT"NORMAL PROBABILITY PLOT OF:"
5030 PRINT"   (1) Every data point
5040 PRINT"   (2) Every other data point
5050 PRINT"   (3) Every  Nth data point
5060 PRINT @645,"NUMBER OF OPTION DESIRED ";CHR$(123);CHR$(176);CHR$(125)
5070 ZI=INKEY$ : IF ZI="" GOTO 5070ELSE IO=VAL(ZI)
5080 PRINT @671,ZI;: FOR I=1 TO 100 : NEXT
5090 IF IO<1 OR IO>3 THEN PRINT @640, CHR$(250);: PRINT @640,"*** NOT A VALID OPTION ***";: FOR I=1 TO
 1000: NEXT : PRINT @640, CHR$(250);: GOTO 5060
5100 IF IO=1 OR IO=2 J=IO : GOTO 5120
5110 PRINT @709,"";:INPUT"Enter N";J
5120 CLS : PRINT @64,"+3 -";
5130 PRINT @192,"+2 -";
5140 PRINT @320,"+1 -";
5150 PRINT @448," 0 -";
5160 PRINT @576,"-1 -";
5170 PRINT @704,"-2 -";
5180 PRINT @832,"-3 -";
5190 FOR I=1 TO 43
5200 SET (5,I)
5210 NEXT I
5220 FOR I=5 TO 125
5230 SET(I,43)
5240 NEXT I
5250 PRINT @20,"NORMAL PROBABILITY PLOT";
5260 SP=INT(Y(1)) : IF Y(IN) = INT (Y(IN)) THEN EP=Y(IN) ELSE EP=INT(Y(IN))+1
5270 PRINT @962, SP;
5280 I= INT( LOG(ABS(EP))/LOG(10) )
```

```
5290 PRINT @1021-I, STR$(EP);
5300 RG=EP - SP
5310 MP= SP + RG/2  :  IF MP>10 THEN MP=INT(MP)
5320 PRINT @990 - INT(I/2), MP;
5330 I2=1 : MO=0 : VR=1
5340 FOR I=1 TO IN+1 STEP J
5350 IF I>IN THEN I=IN
5360 OV= 125 - INT( (EP - Y(I))/RG * 120)
5370 PR= I/(IN+1)
5380 GOSUB 7000
5390 IF XQ < 0 GOTO 5420
5400 DW = 22 - INT( (XQ/3.5) * 22 )
5410 GOTO 5430
5420 DW= 22 + INT ( (ABS(XQ)/3.5) * 22)
5430 SET(OV,DW)
5440 IF I=IN GOTO 5460
5450 NEXT I
5460 IF INKEY$ <> " " GOTO 5460
5470 RETURN
```

EXPONENTIAL PROBABILITY PLOT

```
6000 REM *** E X P O N E N T I A L   P R O B A B I L I T Y   P L O T ***
6010 CLS
6020 PRINT @64,"6 -"
6030 PRINT @192,"5 -"
6040 PRINT @320,"4 -"
6050 PRINT @448,"3 -"
6060 PRINT @576,"2 -"
6070 PRINT @704,"1 -"
6080 PRINT @832,"0"
6090 FOR I=1 TO 39
6100 SET(3,I)
6110 NEXT I
6120 PRINT @835,STRING$(60,".");
6130 PRINT @20,"EXPONENTIAL PROBABILITY PLOT";
6140 CI=Y(IN)/20
6150 HS = 10 [ (INT (LOG(CI*15)/LOG(10)) -1)
6160 CI= INT (CI/HS) + 1
6170 FOR I=0 TO 19
6180 ZH=STR$(CI*(I+1))
6190 IF CI*(I+1) >=100 AND INT(I/2) = I/2 GOTO 6210
6200 PRINT @898+I*3,ZH;
6210 NEXT I
6220 PRINT @990, "X";
6230 IF HS = 1 THEN GOTO 6240ELSE PRINT " *";HS;
6240 IG=0
6250 IX=IN
6260 FOR I=20 TO 1 STEP -1
6270 IF Y(IX) > I*CI*HS THEN IG=IG+1 : IX=IX-1 : GOTO 6270
6280 IF IG=0 OR IG=IH GOTO 6320
6290 OV=I*6
6300 DW= 40 - INT(-LOG(IG/(IN+1)) * 6)
6310 SET (OV,DW)
6320 IH=IG
6330 NEXT I
6340 PRINT @133,"Press SPACE BAR to continue";
6350 IF INKEY$ <> " " GOTO 6350
6360 RETURN
```

INVERSE NORMAL

```
7000 REM *** I N V E R S E  N O R M A L  S U B R O U T I N E
7010 IF PR=.5 THEN XQ=MO : RETURN
7020 C1=2.515517 : C2=.802853 : C3=.010328 : C4=1. : C5=1.432788 : C6=.189269 : C7=.001308
7030 IF PR>.5 THEN RL=SQR(LOG(C4/((C4-PR)*(C4-PR)))) ELSE RL=SQR(LOG(C4/(PR*PR)))
7040 XQ=RL-((C1+(C2*RL)+(C3*RL*RL))/(C4+(C5*RL)+(C6*RL*RL)+(C7*RL*RL*RL)))
7050 IF PR>.5 XQ=MO+(SQR(VR)*XQ) ELSE XQ=MO-(SQR(VR)*XQ)
7060 RETURN
```

PROGRAM SELECTION MENU

```
10000 REM *** P R O G R A M  S E L E C T I O N  M E N U ***
10010 CLS
10020 PRINT TAB(20)"PROGRAM SELECTION MENU"
10030 PRINT @128,"#1  Data I/O, Transformations, Prob. & Inv. Prob. Distributions":PRINT
10040 PRINT"#2  Sample Stat's, 7-Num-Sum, Plots":PRINT
10050 PRINT"#3  Parametric Tests of Hypotheses & Confidence Intervals":PRINT
10060 PRINT"#4  Robust Estimates and Confidence Intervals":PRINT
10070 PRINT"#5  Multiple Linear Regression":PRINT
10080 PRINT"#6  Analysis of Variance"
10090 PRINT @900,"NUMBER OF PROGRAM DESIRED ";CHR$(123);CHR$(176);CHR$(125)
10100 ZI=INKEY$ : IF ZI="" GOTO 10100ELSE IO=VAL(ZI)
10110 PRINT @927,ZI;: FOR I=1 TO 100: NEXT
10120 IF IO<1 OR IO>6 THEN PRINT @896,CHR$(253);:PRINT @896,"***  NOT A VALID OPTION  ***";: FOR I=1 T
O 1000 : NEXT : PRINT @896,CHR$(253);: GOTO 10090
10130 IF IO=2 GOTO 10
10140 IF IO=1 THEN ZT="1" : GOTO 10160
10150 IF IO=3 OR IO=4 THEN ZT="2" ELSE ZT="3"
10160 CLS:PRINT"INSERT TAPE #";ZT;" INTO ESF-80 AND PRESS SPACE BAR"
10170 IF INKEY$ <> " " GOTO 10170
10180 IF IO=1 OR IO=3 OR IO=5 THEN IO=1 ELSE IO=2
10190 @LOAD IO
```

END OF PROGRAM #2

PROGRAM #3

MENU

```
10 REM  *** M A I N   M E N U ***
20 L$=CHR$(123) : R$=CHR$(125) : BL$=CHR$(252)
30 CLS  : PRINT TAB(20) "***  M E N U  ***" : PRINT
40 PRINT"TESTS OF HYPOTHESES & CONFIDENCE INTERVALS FOR DATA ASSUMED:"
50 PRINT @256," (1) Normal (or Central Limit Theorem Applies)
60 PRINT"  (2) Exponential
70 PRINT"  (3) Bernoulli
80 PRINT : PRINT"OTHER OPTIONS:"
90 PRINT : PRINT"  (4) Input Summary Statistics
100 PRINT"  (5) Return to Program Selection Menu
110 JP=696 : KS=5 : GOSUB 760
120 IF IO=2 AND XB<=0 GOTO 150ELSE IF IO=3 AND XB<0 OR IO=3 AND XB>1 GOTO 150
130 ON IO GOSUB 1000, 4000, 6000, 300, 22000
140 GOTO 10
150 CLS
160 IF IO=2 THEN PRINT"EXPONENTIAL PARAMETERS MUST BE GREATER THAN ZERO" : PRINT :  PRINT"THE
 SAMPLE MEAN = ";XB;"AND THEREFORE DOES NOT CORRESPOND TO A" : PRINT"EXPONENTIAL POPULATION
":GOTO 180
170 PRINT"THE SAMPLE MEAN FROM A BERNOULLI POPULATION MUST BE A NUMBER   BETWEEN 0 & 1": PR
INT : PRINT"THE SAMPLE MEAN = ";XB;"AND THEREFORE DOES NOT CORRESPOND TO A" : PRINT"BERNOULL
I POPULATION"
180 PRINT @896,"PRESS SPACE BAR TO RETURN TO MAIN MENU"
190 IF INKEY$<>" " GOTO 190
200 GOTO 10
```

INPUT SUMMARY STATISTICS

```
300 REM *** I N P U T   S U M M A R Y   S T A T I S T I C S ***
310 CLS : I4=1
320 PRINT TAB(20)"INPUT SUMMARY STATISTICS"
330 PRINT @128,"  If summary statistics are entered then any data which was   previously entered  and all samp
le statistics calculated for   such data will be cleared."
340 PRINT : PRINT"  Do you wish to proceed?"
350 T1$="Y" : T2$="N" : JP=448 : GOSUB 710
360 IF ZQ="n" OR ZQ="N" THEN I4=0 : RETURN
370 CLS : PRINT"CLEARING DATA..."
380 FOR I=1 TO 200 : Y(I)=0 : X(I)=0 : NEXT : IN=0 : V1=0 : V=0 : XB=0 : CLS
390 PRINT TAB(20)"INPUT SUMMARY STATISTICS"
400 PRINT : INPUT"Enter Sample Size";IN : PRINT @128, BL$
410 PRINT @128,"Is data assumed to be from an Exponential Population ?"
420 T1$="Y" : T2$="N" : JP=256 : GOSUB 710: IF ZQ="n" OR ZQ="N" GOTO 500
430 PRINT @128, BL$ : PRINT @128,"Will you enter the MEAN or the Exponential Parameter LAMBDA?"
440 T1$="M" : T2$="L" : GOSUB 710
450 PRINT @128, BL$ : PRINT BL$ : PRINT @128,"Enter ";: IF ZQ="m" OR ZQ="M" PRINT "Mean"; ELSE PRINT "La
mbda";
460 INPUT XB : IF XB<=0 THEN PRINT @64,"*** ERROR ***  EXPONENTIAL PARAMETERS MUST BE GREATER
THAN ZERO":GOTO 450
470 IF ZQ="l" OR ZQ="L" THEN XB=1/XB
480 V1=XB*XB : V=V1
```

```
490 RETURN
500 PRINT @128, BL$ : PRINT @128,"Is the sample assumed to be from a Bernoulli Population?"
510 JP=256 : T1$="Y" : T2$="N" : GOSUB 710
520 IF ZQ="n" OR ZQ="N" GOTO 560
530 PRINT @128, BL$ : PRINT BL$ : PRINT @128,"Enter the number of occurrences out of";IN;"trials";:INPUT XB
540 IF XB<0 OR XB>IN GOTO 550ELSE IF XB<>INT(XB) GOTO 550ELSE XB=XB/IN : V1=XB*(1-XB) : V=V1 : RET
URN
550 PRINT @128, BL$ : PRINT @64, "*** ERROR *** ENTER AN INTEGER FROM 0 TO ";IN : GOTO 530
560 PRINT @128, BL$ : PRINT BL$ : PRINT @128,"Enter Sample Mean";:INPUT XB
570 PRINT @256,"Do you want to enter Sample VARIANCE or STD. DEVIATION?"
580 JP=320 : T1$="V" : T2$="D" : GOSUB 710
590 PRINT @256, BL$ : PRINT BL$
600 IF ZQ="v" OR ZQ="V" PRINT @256,"Enter Sample Variance";:INPUT V1 : V=((IN-1)/IN)*V1 : RETURN
610 PRINT @256,"Enter Std. Deviation";:INPUT V1 : V1=V1*V1 : V=((IN-1)/IN)*V1 : RETURN
```

<u>OPTION SELECTION</u>

```
700 REM *** O P T I O N   S E L E C T I O N ***
710 PRINT @JP+5,L$;" ";T1$;" or ";T2$;" ";R$;" ";L$;CHR$(176);R$
720 ZQ=INKEY$ : IF ZQ="" GOTO 720
730 PRINT @JP+18, ZQ : FOR IJ=1 TO 100 : NEXT
740 IF ZQ<>T1$ AND ZQ<>CHR$(ASC(T1$)+32) AND ZQ<>T2$ AND ZQ<>CHR$(ASC(T2$)+32) THEN GOTO 710
750 RETURN
760 PRINT @JP+5,"NUMBER OF OPTION DESIRED ";L$;CHR$(176);R$
770 ZI=INKEY$ : IF ZI="" GOTO 770ELSE IO=VAL(ZI)
780 PRINT @JP+31, ZI; : FOR I=1 TO 100 : NEXT
790 IF IO<1 OR IO>KS THEN PRINT @JP, BL$ : PRINT @JP,"*** NOT A VALID OPTION ***";: FOR I=1 TO 1000 :
 NEXT : PRINT @JP, BL$ : GOTO 760
800 RETURN
```

<u>MENU OF TESTS FOR NORMAL DATA</u>

```
1000 REM *** M E N U   O F   T E S T S   F O R   D A T A   F R O M   N O R M A L   D I S T R I B U T I O N *
**
1010 CLS : PRINT"DATA ENTERED IS ASSUMED TO BE NORMAL
1020 PRINT : PRINT"MENU:"
1030 PRINT"  (1)  Test Hypothesis Concerning the Mean
1040 PRINT"  (2)  Test Hypothesis Concerning the Variance
1050 PRINT"  (3)  Confidence Interval for the Mean
1060 PRINT"  (4)  Confidence Interval for the Variance
1070 JP=640 : KS=4 : GOSUB 760
1080 ON IO GOSUB 1100, 2000, 3000, 3300
1090 RETURN
```

<u>HYPOTHESES CONCERNING THE MEAN</u>

```
1100 REM *** H Y P O T H E S E S   C O N C E R N I N G   M E A N ***
1110 CLS
1120 PRINT @64,"HYPOTHESIS:          NULL                ALTERNATIVE" : PRINT
1130 PRINT"   (1)      mu =  specified value    mu =/= specified value
1140 PRINT"   (2)      mu <= specified value    mu > specified value
1150 PRINT"   (3)      mu >= specified value    mu < specified value
1160 JP=640 : KS=3 : GOSUB 760
1170 CLS : PRINT"Is the variance known?"
1180 JP=64 : T1$="Y" : T2$="N" : GOSUB 710
1190 IF ZQ="n" OR ZQ="N" GOTO 1250
1200 PRINT"Do you want to enter VARIANCE or STD. DEVIATION?"
1210 JP=192 : T1$="V" : T2$="D" : GOSUB 710: ZV=ZQ
1220 PRINT"Enter "; : IF ZV="V" OR ZV="v" PRINT "VARIANCE"; ELSE PRINT"STD. DEVIATION";
```

```
1230 INPUT AV
1240 IF ZV="D" OR ZV="d" AV=AV*AV
1250 CLS : INPUT"Enter the value that the mean is hypothesized to be";MZ
1260 IF ZQ="N" OR ZQ="n" GOTO 1400ELSE ON IO GOSUB 1280, 1480, 1630
1270 RETURN
```

NULL HYPOTHESIS 1

```
1280 REM *** N U L L  H Y P O T H E S I S  1 ***
1290 TZ=( SQR(IN) * (XB - MZ) )/SQR(AV)
1300 PRINT : PRINT"Test Statistic =";TZ
1310 NX=ABS(TZ) : VAR=1 : MU=0 : I2=1 : GOSUB 10000
1320 PRINT"P-Value = ";2*GOX
1330 PRINT: INPUT"Enter Desired Alpha Level";AL
1340 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 1330
1350 PRINT : PRINT"For Alpha =";AL;
1360 IF AL<(2*GOX) THEN PRINT" accept"; ELSE PRINT" reject";
1370 PRINT" the hypothesis that the true mean =";MZ
1380 PRINT"-vs- the alternative hypothesis that true mean =/=";MZ
1390 GOTO 1450
1400 ON IO GOSUB 14;0, 1590, 1750: RETURN
1410 TZ=( SQR(IN) * (XB - MZ) )/SQR(V1)
1420 PRINT : PRINT"Test Statistic = ";TZ
1430 TX=ABS(TZ) : DF=IN-1 : I2=1 : GOSUB 15000
1440 GOTO 1320
1450 PRINT @960,"Press SPACE BAR when ready to return to main menu";
1460 IF INKEY$ <> " " GOTO 1460
1470 RETURN
```

NULL HYPOTHESIS 2

```
1480 REM *** N U L L  H Y P O T H E S I S  2 ***
1490 TZ= (SQR(IN) * (XB - MZ))/SQR(AV)
1500 PRINT: PRINT"Test Statistic =";TZ
1510 NX=TZ : VAR=1 : MU=0 : I2=1 : GOSUB 10000
1520 PRINT"P-Value =";GOX
1530 PRINT:INPUT"Enter desired Alpha Level";AL.
1540 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 1530
1550 PRINT:PRINT"For Alpha =";AL;
1560 IF AL < GOX THEN PRINT"accept"; ELSE PRINT"reject";
1570 PRINT" the hypothesis that the true mean <=";MZ : PRINT "-vs- the alternative hypothesis that the true me
an is >";MZ
1580 GOTO 1450
1590 TZ=( SQR(IN) * (XB-MZ) )/SQR(V1)
1600 PRINT : PRINT"Test Statistic =";TZ
1610 TX=TZ : DF=(IN-1) : I2=1 : GOSUB 15000
1620 GOTO 1520
```

NULL HYPOTHESIS 3

```
1630 REM *** N U L L  H Y P O T H E S I S  3 ***
1640 TZ=(SQR(IN) * (XB-MZ))/SQR(AV)
1650 PRINT : PRINT"Test Statistic =";TZ
1660 NX=TZ : VAR=1 : MU=0 : I2=1 : GOSUB 10000
1670 PRINT"P-Value =";FOX
1680 PRINT : INPUT"Enter desired Alpha Level";AL
1690 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 1680
1700 PRINT : PRINT"For Alpha =";AL;
1710 IF AL < FOX THEN PRINT"accept"; ELSE PRINT "reject";
```

```
1720 PRINT" the hypothesis that the true mean >=";MZ
1730 PRINT "-vs- the alternative hypothesis that true mean <";MZ
1740 GOTO 1450
1750 TZ=( SQR(IN) * (XB-MZ) )/SQR(V1)
1760 PRINT : PRINT"Test Statistic =";TZ
1770 TX=TZ : DF=IN-1 : I2=1 : GOSUB 15000
1780 GOTO 1670
```

HYPOTHESES CONCERNING THE VARIANCE

```
2000 REM *** H Y P O T H E S E S  C O N C E R N I N G  V A R I A N C E ***
2010 CLS
2020 PRINT @64,"HYPOTHESIS:       NULL              ALTERNATIVE" : PRINT
2030 PRINT"   (1)  Var = specified value    Var =/= specified value
2040 PRINT"    or  Std. Dev.  -dito-        Std. Dev.  -dito-" : PRINT
2050 PRINT"   (2)  Var <= specified value   Var > specified value
2060 PRINT"    or  Std. Dev.  -dito-        Std. Dev.  -dito-" : PRINT
2070 PRINT"   (3)  Var >= specified value   Var < specified value
2080 PRINT"    or  Std. Dev.  -dito-        Std. Dev.  -dito-": PRINT
2090 JP=896 : KS=3 : GOSUB 760
2100 CLS : PRINT"Do you want to enter VARIANCE or STD. DEVIATION?"
2110 JP=64 : T1$="V" : T2$="D" : GOSUB 710: ZV=ZQ
2120 PRINT"Enter the value the ";: IF ZV="V" OR ZV="v" PRINT"VARIANCE"; ELSE PRINT"STD. DEVIATION";
2130 PRINT" is hypothesized to be";: INPUT AV
2140 IF ZV="D" OR ZV="d" AV=AV*AV
2150 PRINT @192,"Is the mean known?"
2160 JP=256 : T1$="Y" : T2$="N" : GOSUB 710
2170 IF ZQ="n" OR ZQ="N" GOTO 2190
2180 INPUT"Enter Mean";MZ
2190 IF ZQ="N" OR ZQ="n"  OR I4=1 TZ=((IN-1) * V1)/AV : GOTO 2210
2200 TZ=0 : FOR I=1 TO IN : TZ=TZ + (Y(I) - MZ)[2 : NEXT I : TZ=TZ/AV
2210 PRINT : PRINT "Test Statistic =";TZ
2220 IF ZQ="N" DF=IN-1 ELSE DF=IN
2230 CX=TZ : I2=1 : GOSUB 12000
2240 PRINT "P-Value =";
2250 ON IO GOSUB 2270, 2410, 2520
2260 RETURN
```

NULL HYPOTHESIS 1

```
2270 REM *** N U L L  H Y P O T H E S I S  1 ***
2280 IF 2*FOX <=1 PRINT 2*FOX ELSE PRINT 2*QOX
2290 PRINT : INPUT"Enter Desired Alpha Level";AL
2300 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 2290
2310 PRINT : PRINT"For Alpha =";AL
2320 IF 2*FOX > 1 GOTO 2350
2330 IF AL < 2*FOX  PRINT"Accept"; ELSE PRINT"Reject";
2340 GOTO 2360
2350 IF AL < 2*QOX PRINT"Accept"; ELSE PRINT"Reject";
2360 PRINT" the hypothesis that ";: IF ZV="V" OR ZV="v" PRINT "VARIANCE"; ELSE PRINT "STD. DEVIATION";
2370 PRINT " =";:IF ZV="V" OR ZV="v" PRINT AV ELSE PRINT AV[.5
2380 PRINT"-vs- the alternative that ";: IF ZV="V" OR ZV="v" PRINT"VARIANCE"; ELSE PRINT "STD. DEVIATIO
N";
2390 PRINT" =/=";:IF ZV="V" OR ZV="v" PRINT AV ELSE PRINT AV[.5
2400 GOTO 1450
```

NULL HYPOTHESIS 2

```
2410 REM *** N U L L  H Y P O T H E S I S  2 ***
2420 PRINT QOX
2430 PRINT : INPUT"Enter Desired Alpha Level";AL
2440 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 2430
2450 PRINT : PRINT "For Alpha =";AL
2460 IF AL<QOX PRINT "Accept"; ELSE PRINT "Reject";
2470 PRINT" the hypothesis that the ";: IF ZV="V" OR ZV="v" PRINT"VARIANCE"; ELSE PRINT"STD. DEVIATION
";
2480 PRINT" <=";: IF ZV="V" OR ZV="v" PRINT AV ELSE PRINT AV[.5
2490 PRINT"-vs- the alternative that the ";: IF ZV="V" OR ZV="v" PRINT"VARIANCE"; ELSE PRINT"STD. DEVIA
TION";
2500 PRINT" >";: IF ZV="V" OR ZV="v" PRINT AV ELSE PRINT AV[.5
2510 GOTO 1450
```

NULL HYPOTHESIS 3

```
2520 REM *** N U L L  H Y P O T H E S I S  3 ***
2530 PRINT FOX
2540 PRINT : INPUT"Enter Desired Alpha Level";AL
2550 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 2540
2560 PRINT : PRINT"For Alpha =";AL
2570 IF AL<FOX PRINT "Accept"; ELSE PRINT "Reject";
2580 PRINT" the hypothesis that the ";: IF ZV="V" OR ZV="v" PRINT"VARIANCE"; ELSE PRINT"STD. DEVIATION
";
2590 PRINT" >=";: IF ZV="V" OR ZV="v" PRINT AV ELSE PRINT AV[.5
2600 PRINT"-vs- the alternative that the ";: IF ZV="V" OR ZV="v" PRINT "VARIANCE"; ELSE PRINT"STD. DEVIA
TION";
2610 PRINT " <";: IF ZV="V" OR ZV="v" PRINT AV ELSE PRINT AV[.5
2620 GOTO 1450
```

CONFIDENCE INTERVAL FOR THE MEAN

```
3000 REM *** C O N F I D E N C E  I N T E R V A L  F O R  T H E  M E A N ***
3010 CLS
3020 PRINT"Is Variance Known?" : JP=64 : T1$="Y" : T2$="N" : GOSUB 710
3030 IF ZQ="N" OR ZQ="n" GOTO 3190
3040 PRINT @0, BL$ : PRINT BL$ : PRINT @0,"Do you want to enter VARIANCE or STD. DEVIATION?" : T1$="V"
: T2$="D" : JP=64 : GOSUB 710: ZV=ZQ
3050 PRINT"Enter ";: IF ZV="V" OR ZV="v" PRINT"VARIANCE"; ELSE PRINT "STD. DEVIATION";
3060 INPUT AV
3070 IF ZV="d" OR ZV="D" AV=AV*AV
3080 PRINT : INPUT "Enter Desired Confidence Level (e.g. .95)";AL : IF AL>1 THEN AL=AL/100
3090 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 3080
3100 MO=0 : VR=1 : PR=(1+AL)/2 : I2=1 : GOSUB 17000
3110 CLS : PRINT @84, AL*100;"% CONFIDENCE LIMITS"
3120 PRINT : PRINT "TWO SIDED CONFIDENCE LIMITS:"
3130 PRINT"     ";:PRINT USING"#####.###";XB-(XQ * SQR(AV/IN));:PRINT"  < True Mean <";:PRINT USING"#
####.###";XB+(XQ * SQR(AV/IN))
3140 PR=AL : GOSUB 17000
3150 PRINT : PRINT"ONE SIDED CONFIDENCE LIMITS:"
3160 PRINT"   Upper =";:PRINT USING"#####.###";XB + (XQ * SQR(AV/IN))
3170 PRINT"   Lower =";:PRINT USING"#####.###";XB - (XQ * SQR(AV/IN))
3180 GOTO 1450
3190 PRINT : INPUT "Enter Desired Confidence Level (e.g. .95)";AL : IF AL>1 THEN AL=AL/100
3200 IF AL<0  OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 3190
3210 DF=(IN-1) : PR=(1+AL)/2 : I2=1 : GOSUB 19000
3220 CLS : PRINT @84, AL*100;"% CONFIDENCE LIMITS"
3230 PRINT : PRINT "TWO SIDED CONFIDENCE LIMITS:"
3240 PRINT"     ";:PRINT USING"#####.###";XB-(XQ * SQR(V/(IN-1)));:PRINT"  < True Mean <";:PRINT USING"
#####.###";XB+(XQ * SQR(V/(IN-1)))
3250 PR=AL : GOSUB 19000
```

```
3260 PRINT : PRINT"ONE SIDED CONFIDENCE LIMITS:"
3270 PRINT"   Upper =";:PRINT USING"#####.###";XB + (XQ * SQR(V/(IN-1)))
3280 PRINT"   Lower =";:PRINT USING"#####.###";XB - (XQ * SQR(V/(IN-1)))
3290 GOTO 1450
```

CONFIDENCE INTERVAL FOR THE VARIANCE

```
3300 REM *** C O N F I D E N C E   I N T E R V A L   F O R   V A R I A N C E ***
3310 CLS
3320 PRINT"Is the Mean Known?" : JP=64 : T1$="Y" : T2$="N" : GOSUB 710
3330 IF ZQ="N" OR ZQ="n" GOTO 3540
3340 INPUT"Enter Mean";XB
3350 INPUT"Enter Desired Confidence Level (e.g. .95)";AL : IF AL>1 THEN AL=AL/100
3360 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 3350
3370 IF I4=1 W2=IN*V : GOTO 3390
3380 W2=0 : FOR I=1 TO IN : W2=W2 + ( Y(I) - XB )[2 : NEXT I
3390 CLS : PRINT @84, AL*100;"% CONFIDENCE LIMITS"
3400 PRINT @320,"TWO SIDED CONFIDENCE LIMITS:"
3410 DF=IN : PR=(1+AL)/2 : I2=1 : GOSUB 18000
3420 PRINT@384,"     ";:PRINT USING"#####.###";W2/XQ;:PRINT"  < VARIANCE <"
3430 PRINT@448,"     ";:PRINT USING"#####.###";(W2/XQ)[.5;:PRINT"  < STD. DEVIATION <"
3440 PR=(1-AL)/2 : GOSUB 18000
3450 PRINT @416,"";:PRINT USING"#####.###";W2/XQ;
3460 PRINT @483,"";:PRINT USING"#####.###";(W2/XQ)[.5
3470 PRINT:PRINT"ONE SIDED CONFIDENCE LIMITS:"
3480 PRINT"          Variance       Std. Dev."
3490 PR=1-AL : GOSUB 18000
3500 PRINT "  Upper =";:PRINT USING"#####.###";W2/XQ;: PRINT USING"############.###";(W2/XQ)[.5
3510 PR=AL : GOSUB 18000
3520 PRINT "  Lower =";:PRINT USING"#####.###";W2/XQ;: PRINT USING"############.###";(W2/XQ)[.5
3530 GOTO 1450
3540 PRINT : INPUT"Enter desired confidence level (e.g. .95)";AL : IF AL>1 THEN AL=AL/100
3550 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 3540
3560 CLS : PRINT @84, AL*100;"% CONFIDENCE LIMITS"
3570 PRINT @320,"TWO SIDED CONFIDENCE LIMITS:"
3580 DF=(IN-1) : PR=(1+AL)/2 : I2=1 : GOSUB 18000
3590 PRINT@384,"     ";:PRINT USING"#####.###";(IN*V)/XQ;:PRINT"  < VARIANCE <"
3600 PRINT@448,"     ";:PRINT USING"#####.###";((IN*V)/XQ)[.5;:PRINT"  < STD. DEVIATION <";
3610 PR=(1-AL)/2 : GOSUB 18000
3620 PRINT @416,"";:PRINT USING"#####.###";(IN*V)/XQ;
3630 PRINT @483,"";:PRINT USING"#####.###";((IN*V)/XQ)[.5
3640 PRINT:PRINT"ONE SIDED CONFIDENCE LIMITS:"
3650 PRINT"          Variance       Std. Dev."
3660 PR=1-AL : GOSUB 18000
3670 PRINT"  Upper =";:PRINT USING"#####.###";(IN*V)/XQ;:PRINT USING"############.###";((IN*V)/X
Q)[.5
3680 PR=AL : GOSUB 18000
3690 PRINT"  Lower =";:PRINT USING"#####.###";(IN*V)/XQ;:PRINT USING"############.###";((IN*V)/X
Q)[.5
3700 GOTO 1450
```

MENU OF TESTS FOR EXPONENTIAL DATA

```
4000 REM *** M E N U   O F   T E S T S   F O R   E X P O N E N T I A L   D A T A ***
4010 CLS : PRINT "DATA IS ASSUMED TO BE EXPONENTIAL
4020 PRINT : PRINT"MENU:"
4030 PRINT"  (1) Test Hypothesis Concerning Mean (or 1/mean)
4040 PRINT"  (2) Confidence Interval for Mean (or 1/mean)
4050 PRINT : PRINT"NOTE: both of the above options allow either the Mean or the"
4060 PRINT"     exponential parameter Lambda (1/mean) to be used as the" : PRINT "     argument
4070 JP=896 : XS=2 : GOSUB 760
4080 ON IO GOSUB 4100, 5000
4090 RETURN
```

HYPOTHESES CONCERNING THE MEAN

```
4100 REM *** H Y P O T H E S E S   C O N C E R N I N G   M E A N ***
4110 CLS : PRINT TAB(28)"MENU:"
4120 PRINT @128,"Do you want to use the Mean or Lambda (1/mean) as the argument?"
4130 JP=192 : T1$="M" : T2$="L" : GOSUB 710: ZA=ZQ
4140 IF ZA="1" OR ZA="L" THEN A$="lambda" ELSE A$="mean"
4150 PRINT @128, CHR$(255) : PRINT CHR$(255)
4160 PRINT @64,"HYPOTHESIS:"
4170 PRINT @128,"            NULL                    ALTERNATIVE"
4180 PRINT"(1) ";A$;" = specified value";: PRINT TAB(35) A$;" =/= specified value"
4190 PRINT"(2) ";A$;" <= specified value";: PRINT TAB(35) A$;" > specified value"
4200 PRINT"(3) ";A$;" >= specified value";: PRINT TAB(35) A$;" < specified value"
4210 JP=896 : KS=3 : GOSUB 760
4220 CLS : PRINT "Enter the hypothesized value of ";A$;:INPUT MZ
4230 IF ZA="1" OR ZA="L" THEN CX=2*IN*XB*MZ  ELSE CX=2*IN*XB/MZ
4240 DF=2*IN
4250 PRINT : PRINT "Test Statistic =";CX
4260 GOSUB 12000
4270 ON IQ GOSUB 4290, 4410, 4530
4280 RETURN
```

NULL HYPOTHESIS 1

```
4290 REM *** N U L L   H Y P O T H E S I S   1 ***
4300 IF 2*FOX <= 1 THEN PV=2*FOX ELSE PV=2*QOX
4310 PRINT"P-Value =";PV
4320 PRINT : INPUT"Enter Desired Alpha Level";AL
4330 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO4320
4340 PRINT : PRINT"For Alpha =";AL
4350 IF AL<PV PRINT"Accept"; ELSE PRINT "Reject";
4360 PRINT" the hypothesis that the true ";A$;" =";MZ
4370 PRINT"-vs- the alternative that the true ";A$;" =/=";MZ
4380 PRINT @960,"PRESS SPACE BAR TO RETURN TO MAIN MENU";
4390 IF INKEY$ <> " " GOTO 4390
4400 RETURN
```

NULL HYPOTHESIS 2

```
4410 REM *** N U L L   H Y P O T H E S I S   2 ***
4420 PRINT"P-Value =";: IF ZA="1" OR ZA="L" PRINT FOX ELSE PRINT QOX
4430 PRINT : INPUT"Enter Desired Alpha Level";AL
4440 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 4430
4450 PRINT"For Alpha =";AL
4460 IF ZA="1" OR ZA="L" GOTO 4490
4470 IF AL<QOX PRINT "Accept"; ELSE PRINT "Reject";
4480 GOTO 4500
4490 IF AL<FOX PRINT "Accept"; ELSE PRINT "Reject";
4500 PRINT " the hypothesis that the true ";A$;" <=";MZ
4510 PRINT "-vs- the alternative that the true ";A$;" >";MZ
4520 GOTO 4380
```

NULL HYPOTHESIS 3

```
4530 REM *** N U L L   H Y P O T H E S I S   3 ***
4540 PRINT"P-Value =";: IF ZA="1" OR ZA="L" PRINT QOX ELSE PRINT FOX
4550 PRINT : INPUT"Enter Desired Alpha Level";AL
4560 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 4550
4570 PRINT "For Alpha =";AL
```

```
4580 IF ZA="1" OR ZA="L" GOTO 4610
4590 IF AL<FOX PRINT "Accept"; ELSE PRINT "Reject";
4600 GOTO 4620
4610 IF AL<OOX PRINT "Accept"; ELSE PRINT "Reject";
4620 PRINT " the hypothes that the true ";A$;" >=";MZ
4630 PRINT " -vs- the alternative that the true ";A$;" <";MZ
4640 GOTO 4380
```

CONFIDENCE INTERVAL FOR THE EXPONENTIAL MEAN

```
5000 REM *** C O N F I D E N C E  L E V E L  F O R  E X P O N E N T I A L  M E A N ***
5010 CLS : PRINT"CONFIDENCE INTERVAL FOR EXPONENTIAL MEAN"
5020 PRINT : INPUT"Enter Desired Confidence Level ( e.g. .95 )";AL : IF AL>1 THEN AL=AL/100
5030 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 5020
5040 PR=(1-AL)/2 : DF=2*IN : I2=1 : GOSUB 18000
5050 L=XQ/(2*IN*XB)
5060 PR=(1+AL)/2 : GOSUB 18000
5070 U=XQ/(2*IN*XB)
5080 CLS : PRINT @34, AL*100;"% CONFIDENCE LIMITS"
5090 PRINT : PRINT"TWO SIDED CONFIDENCE LIMITS:"
5100 PRINT USING"######.###";1/U;:PRINT" < True Mean <";:PRINT USING"######.###";1/L
5110 PRINT USING"######.###";L;:PRINT" <  Lambda  <";:PRINT USING"######.###";U
5120 PRINT : PRINT"ONE SIDED CONFIDENCE LIMITS:"
5130 PR=AL : GOSUB 18000: U=XQ/(2*IN*XB)
5140 PR=1-AL : GOSUB 18000: L=XQ/(2*IN*XB)
5150 PRINT "MEAN:   Upper =";1/L
5160 PRINT "        Lower =";1/U
5170 PRINT "LAMBDA: Upper =";U
5180 PRINT "        Lower =";L
5190 GOTO 4380
```

MENU OF TESTS FOR BERNOULLI DATA

```
6000 REM *** M E N U  O F  T E S T S  F O R  D A T A  A S S U M E D  T O  B E  B E R N O U L L I
6010 CLS : PRINT"DATA IS ASSUMED TO BE BERNOULLI
6020 PRINT : PRINT"MENU:"
6030 PRINT"   (1)  Test Hypothesis Concerning Proportion (P)
6040 PRINT"   (2)  Confidence Interval for P
6050 JP=448 : KS=2 : GOSUB 760
6060 ON IO GOSUB 6100, 7000
6070 RETURN
```

HYPOTHESES CONCERNING THE PROPORTION

```
6100 REM *** H Y P O T H E S E S  C O N C E R N I N G  P R O P O R T I O N ***
6110 CLS : PRINT @64,"HYPOTHESIS:        NULL                ALTERNATIVE" : PRINT
6120 PRINT"    (1)     P =  specified value     P =/= specified value
6130 PRINT"    (2)     P <= specified value     P > specified value
6140 PRINT"    (3)     P >= specified value     P < specified value
6150 JP=640 : KS=3 : GOSUB 760
6160 CLS : INPUT"Enter Hypothesized Value of  P";PB
6170 IF PB<0 OR PB>1 THEN PRINT"*** ERROR ***   P  MUST BE A VALUE BETWEEN  0  AND  1" : FOR I=1 TO
2000 : NEXT : GOTO 6160
6180 IF IN >= 20 THEN YS=(IN * XB - IN * PB) / SQR(IN * PB * (1-PB)) ELSE YS=IN * XB
6190 PRINT : PRINT"Test Statistic =";YS; : IF IN < 20 PRINT"     (Using Binomial CDF)" ELSE PRINT"     (Using N
ormal CDF)"
6200 ON IO GOSUB 6220, 6410, 6550
6210 RETURN
```

NULL HYPOTHESIS 1

```
6220 REM *** N U L L  H Y P O T H E S I S  1 ***
6230 IF IN >= 20 GOTO 6300
6240 N=IN : I2=1 : K=YS : GOSUB 11000
6250 PV=2*PR
6260 IF PV < 1 GOTO 6290
6270 'K=YS-1 : GOSUB 12000
6280 PV=2*(1-PR)
6290 GOTO 6320
6300 MU=0 : VAR=1 : NX=ABS(YS) : I2=1 : GOSUB 10000
6310 PV=2*QOX
6320 PRINT : PRINT"P-Value =";PV
6330 PRINT : INPUT"Enter Desired Alpha Level";AL
6340 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1" : GOTO 6330
6350 IF AL < PV PRINT "Accept"; ELSE PRINT "Reject";
6360 PRINT" the hypothesis that the true value of  P =";PB
6370 PRINT"-vs- the alternative that the true value of  P =/=";PB
6380 PRINT : PRINT"***  PRESS SPACE BAR TO RETURN TO MAIN MENU"
6390 IF INKEY$ <> " " GOTO 6390
6400 RETURN
```

NULL HYPOTHESIS 2

```
6410 REM *** N U L L  H Y P O T H E S I S  2 ***
6420 IF IN >= 20 GOTO 6460
6430 N=IN : K=YS-1 : I2=1 : GOSUB 11000
6440 PV=1-PR
6450 GOTO 6480
6460 MU=0 : VAR=1 : NX=YS : I2=1 : GOSUB 10000
6470 PV=QOX
6480 PRINT : PRINT"P-Value =";PV
6490 PRINT : INPUT"Enter Desired Alpha Level";AL
6500 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 6490
6510 IF AL < PV PRINT "Accept"; ELSE PRINT "Reject";
6520 PRINT" the hypothesis that the true value of  P <=";PB
6530 PRINT"-vs- the alternative that the true value of  P >";PB
6540 GOTO 6380
```

NULL HYPOTHESIS 3

```
6550 REM *** N U L L  H Y P O T H E S I S  3 ***
6560 IF IN >= 20 GOTO 6600
6570 N=IN : K=YS : I2=1 : GOSUB 11000
6580 PV=PR
6590 GOTO 6620
6600 MU=0 : VAR=1 : NX=YS : I2=1 : GOSUB 10000
6610 PV=FOX
6620 PRINT : PRINT"P-Value =";PV
6630 PRINT : INPUT"Enter Desired Alpha Level";AL
6640 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1" : GOTO 6630
6650 IF AL < PV PRINT "Accept"; ELSE PRINT "Reject";
6660 PRINT" The hypothesis that the true value of  P >=";PB
6670 PRINT"-vs- the alternative that the true value of  P <";PB
6680 GOTO 6380
```

CONFIDENCE INTERVAL FOR PROPORTION

```
7000 REM *** C O N F I D E N C E   I N T E R V A L   F O R   P R O P O R T I O N ***
7010 CLS
7020 INPUT"Enter the desired confidence level (e.g. .95)";AL : IF AL>1 AL=AL/100
7030 IF AL<0 OR AL>1 THEN PRINT"*** ERROR ***  ENTER A VALUE BETWEEN  0 & 1":GOTO 7020
7040 PR=(1+AL)/2
7050 DN=2*IN - 2*IN*XB + 2 : DD=2*IN*XB : I2=1 : GOSUB 20000
7060 CLS : PRINT @84, AL*100;"% CONFIDENCE LIMITS"
7070 PRINT : PRINT"TWO SIDED CONFIDENCE LIMITS:"
7080 L=(IN * XB) / ((IN*XB) + (IN - IN*XB + 1) * XQ)
7090 PRINT "     ";L;"< True Value of  P <";
7100 DN=2*IN*XB + 2 : DD=2*IN - 2*IN*XB : GOSUB 20000
7110 U=((IN * XB +1) * XQ) / ((IN - IN*XB) + (IN * XB + 1) * XQ)
7120 PRINT U
7130 PRINT : PRINT"ONE SIDED CONFIDENCE LIMITS:"
7140 PR=AL : GOSUB 20000
7150 PRINT"     Upper =";((IN * XB + 1) * XQ) / ((IN - IN*XB) + (IN*XB + 1) * XQ)
7160 DN=2*IN - 2*IN*XB : DD=2*IN*XB : GOSUB 20000
7170 PRINT"     Lower =";(IN * XB) / (IN * XB + (IN - IN*XB + 1) * XQ)
7180 GOTO 6380
```

NORMAL DISTRIBUTION

```
10000 REM *** N O R M A L   D I S T R I B U T I O N ***
10010 SD=SQR(VAR)
10020 XN=(NX-MU)/SD:AX=ABS(XN)
10030 XZ=.3989423*EXP((-AX*AX)/2)
10040 NT=1/(1+.2316419*AX)
10050 QOX=XZ*((.3193815*NT)-(.3565638*NT[2)+(1.781478*NT[3)-(1.821256*NT[4)+(1.330274*NT[5))
10060 IF XN<0 GOTO 10090
10070 FOX=1-QOX
10080 RETURN
10090 FOX=QOX
10100 QOX=1-FOX
10110 RETURN
```

BINOMIAL DISTRIBUTION

```
11000 REM *** B I N O M I A L   D I S T R I B U T I O N ***
11010 PR=0
11020 IF K<0 GOTO 11080
11030 IF K >= N THEN PR=1 : GOTO 11080
11040 FOR BB=0 TO K
11050 TB=N : GOSUB 11090
11060 PR=PR+ BC * PB[BB * (1-PB)[(N-BB)
11070 NEXT BB
11080 RETURN
11090 BC=0 : B=0 : BT=0 : SB=BB : QB=TB-BB
11100 FOR BI=1 TO TB
11110 IF SB>=QB GOTO 11170
11120 IF SB >= BI THEN B=B + LOG(BI)
11130 IF QB > BI GOTO 11210
11140 IF QB = BI THEN BC=-B : GOTO 11210
11150 BC=BC + LOG(BI)
11160 GOTO 11210
11170 IF QB >= BI THEN B=B + LOG(BI)
11180 IF SB > BI THEN GOTO 11210
11190 IF SB = BI THEN BC=-B : GOTO 11210
11200 BC=BC + LOG(BI)
```

```
11210 NEXT BI
11220 BC=EXP(BC)
11230 RETURN
```

CHI-SQUARE DISTRIBUTION

```
12000 REM *** C H I - S Q U A R E  D I S T R I B U T I O N ***
12010 IF DF>30 GOTO 12220
12020 DP=1
12030 FOR I=INT(DF) TO 2 STEP -2
12040 DP=DP*I
12050 NEXT I
12060 NP=CX[(INT((DF+1)/2))*EXP(-CX/2)/DP
12070 IF INT(DF/2)=DF/2 GOTO 12100
12080 PC=SQR(2/CX/3.141593)
12090 GOTO 12110
12100 PC=1
12110 LC=1
12120 MC=1
12130 D=DF
12140 D=D+2
12150 MC=MC*CX/D
12160 IF MC<.000001 GOTO 12190
12170 LC=LC+MC
12180 GOTO 12140
12190 FOX=PC*NP*LC
12200 GOX=1-FOX
12210 RETURN
12220 NX=((CX/DF)[(1/3) - (1-(2/(9*DF))))/SQR(2/(9*DF))
12230 MU=0 : VAR=1 : I2=1
12240 GOSUB 10000
12250 RETURN
```

t - DISTRIBUTION

```
15000 REM *** t - D I S T R I B U T I O N ***
15010 IF TX<0 I3=1 ELSE I3=0
15020 IF DF>25 GOTO 15450
15030 THETA=ATN(ABS(TX)/SQR(DF))
15040 IF DF<>1 GOTO 15090
15050 AT=.6366198*THETA
15060 FOX=(AT+1)/2
15070 GOX=1-FOX
15080 GOTO 15400
15090 IF INT(DF/2)=DF/2 GOTO 15220
15100 IF DF=3 GOTO 15370
15110 TS=0 : TP=1 : TC=COS(THETA)
15120 FOR I=1 TO INT((DF-3)/2)
15130 TP=TP*2*I/(1+2*I)
15140 TC=TC*(COS(THETA))[2)
15150 TS=TS+(TP*TC)
15160 NEXT I
15170 TS=TS+COS(THETA)
15180 AT=.6366198*(THETA+(SIN(THETA)*TS))
15190 FOX=(AT+1)/2
15200 GOX=1-FOX
15210 GOTO 15400
15220 IF DF<>2 GOTO 15270
15230 AT=SIN(THETA)
15240 FOX=(AT+1)/2
15250 GOX=1-FOX
15260 GOTO 15400
```

```
15270 TS=0 : TP=1 : TC=1
15280 FOR I=1 TO INT((DF-2)/2)
15290 TP=TP*((2*I)-1)/(2*I)
15300 TC=TC*(COS(THETA)[2)
15310 TS=TS+(TP*TC)
15320 NEXT I
15330 AT=SIN(THETA)*(1+TS)
15340 FOX=(AT+1)/2
15350 QOX=1-FOX
15360 GOTO 15400
15370 AT=.6366198*(THETA+(SIN(THETA)*COS(THETA)))
15380 FOX=(AT+1)/2
15390 QOX=1-FOX
15400 PX=TX
15410 IF I3=0 RETURN
15420 QOX=FOX
15430 FOX=1-QOX
15440 RETURN
15450 NX=( TX * (1 - .25/DF) )/SQR(1 + .5*TX[2/DF)
15460 MU=0 : VAR=1 : I2=1 : GOSUB 10000
15470 PX=TX
15480 RETURN
```

F - DISTRIBUTION

```
16000 REM *** F - D I S T R I B U T I O N  ***
16010 IF FX=0 THEN FOX=0 : QOX=1 : RETURN
16020 IF DN>40 AND DD>40 GOTO 16610
16030 K1=DN : K2=DD
16040 A=DN*FX/(DN*FX+DD) : A1=1-A
16050 IF A1<1E-37 THEN A1=1E-37
16060 D1=DN*.5 : D2=DD*.5 : D3=D1+D2-1
16070 R=0 : S1=0 : S2=0 : DL=1 : C=.25
16080 XM=1 : XK=1 : PI=3.141593 : KN=DD
16090 KM=INT(D2)*2
16100 IF KM<>KN GOTO 16280
16110 KN=D2-1
16120 IF KN=0 GOTO 16210
16130 FOR I=1 TO KN
16140 S1=DL+S1*R
16150 D2=D2-1
16160 D3=D3-1
16170 TEM=A1/D2
16180 R=D3*TEM
16190 S2=(R+TEM)*S2
16200 NEXT I
16210 S1=DL+S1*R
16220 DL=0
16230 T=-1
16240 D3=-1
16250 S2=A*S2
16260 C=C+.5
16270 GOTO 16460
16280 KN=INT(D2)
16290 IF KN=0 GOTO 16380
16300 FOR I=1 TO KN
16310 S1=DL+S1*R
16320 D2=D2-1
16330 D3=D3-1
16340 TEM=A1/D2
16350 R=D3*TEM
```

```
16360 S2=(R+TEM)*S2
16370 NEXT I
16380 S1=XK*S1
16390 S2=XK*S2
16400 ART=SQR(A1)
16410 XM=XM*ART
16420 T=(XM-ART)/A1
16430 D3=-.5
16440 XK=2/PI
16450 C=C*2
16460 IF C>.875 GOTO 16550
16470 D2=D1
16480 D3=D2+D3
16490 S2=S1
16500 S1=0
16510 A1=A
16520 IF A1<1E-37 THEN A1=1E-37
16530 KN=DN
16540 GOTO 16090
16550 IF C<1.125 THEN DL=4/PI * ATN(T)
16560 FOX=XM*(S2-S1)-DL
16570 IF FOX<0 THEN FOX=0
16580 IF FOX>1 THEN FOX=1
16590 QOX=1-FOX
16600 RETURN
16610 NX=(((FX[(1/3))*(1-(2/(9*DD))))-(1-(2/(9*DN))))/SQR((2/(9*DN))+(FX[(2/3))*2/(9*DD))
16620 MU=0 : VAR=1
16630 GOSUB 10000
16640 RETURN
```

INVERSE NORMAL

```
17000 REM *** I N V E R S E  N O R M A L  D I S T R I B U T I O N ***
17010 IF PR>.5 THEN RL=SQR(LOG(1/((1-PR)[2))) ELSE RL=SQR(LOG( 1/(PR[2)))
17020 XQ=RL-((2.515517+(.802853*RL)+(.010328*RL[2))/(1+(1.432788*RL)+(.189269*RL[2)+(.001308*RL[3)))
17030 IF PR>=.5 THEN XQ=MO+(SQR(VR)*XQ) ELSE XQ=MO-(SQR(VR)*XQ)
17040 RETURN
```

INVERSE CHI-SQUARE

```
18000 REM *** I N V E R S E  C H I - S Q U A R E  D I S T R I B U T I O N ***
18010 MO=0 : VR=1 : I2=1 : PZ=PR
18020 IF DF>1 GOTO 18070
18030 PR=.5*(1-PZ)
18040 GOSUB 17000
18050 XQ=XQ[2 : PR=PZ
18060 RETURN
18070 IF DF>2 GOTO 18100
18080 XQ=-2*LOG(1-PZ)
18090 RETURN
18100 GOSUB 17000
18110 DQ=XQ
18120 IF DF>(2+INT(4*ABS(DQ))) GOTO 18220
18130 X9=1.0000886 - .2237368/DF - .01513904/DF[2
18140 X9=X9+(DF[-.5 * DQ * (.4713941 + .02607083/DF - .008986007/DF[2))
18150 X9=X9 + (1/DF * DQ[2 * (.0001348028 + .01128186/DF + .02277679/DF[2))
18160 X9=X9 + (DF[-1.5 * DQ[3 * (-.008553069 - .01153761/DF - .01323293/DF[2))
18170 X9=X9 + (DF[-2 * DQ[4 * (.00312558 + .005169654/DF - .006950356/DF[2))
18180 X9=X9 + (DF[-2.5 * DQ[5 * (-.0008426812 + .00253001/DF + .001060438/DF[2))
18190 X9=X9 + (DF[-3 * DQ[6 * (.00009780499 - .001450117/DF + .001565326/DF[2))
18200 XQ=DF*X9[3
```

```
18210 RETURN
18220 X9=1 - 2/(9*DF) + (4*DQ[4 + 16*DQ[2 -28)/(1215*DF[2)
18230 X9=X9 + (8*DQ[6 + 720*DQ[4 + 3126*DQ[2 + 2904)/(229635*DF[3)
18240 X8=DQ/3 + (-DQ[3 + 3*DQ)/(162*DF)
18250 X8=X8 - (3*DQ[5 + 40*DQ[3 + 45*DQ)/(5832*DF[2)
18260 X8=X8 + (301*DQ[7 - 1517*DQ[5 -32769*DQ[3 - 79349*DQ)/(7873200*DF[3)
18270 X8= SQR(2/DF) * X8
18280 XQ=DF * ((X8+X9)[3)
18290 RETURN
18300 CX=XQ
18310 DELTA=.01
18320 GOSUB 12000
18330 IF ABS(PR-FOX)<.00001 THEN XQ=CX : RETURN
18340 IF PR < FOX THEN CX=CX-DELTA
18350 IF PR < FOX THEN DELTA=DELTA/10
18360 IF PR < FOX GOTO 18320
18370 CX=CX+DELTA
18380 GOTO 18320
```

INVERSE - t

```
19000 REM *** I N V E R S E  t - D I S T R I B U T I O N ***
19010 MO=0 : VR=1 : I2=1
19020 GOSUB 17000
19030 XC=XQ
19040 XQ=(.25 * (XC[3 + XC))/DF
19050 XQ=XQ + ((1/96) * (5*XC[5 + 16*XC[3 + 3*XC))/DF[2
19060 XQ=XQ + ((1/384) * (3*XC[7 + 19*XC[5 + 17*XC[3 -15*XC))/DF[3
19070 XQ=XQ + ((1/92160) * (79*XC[9 + 776*XC[7 + 1482*XC[5 -1920*XC[3 -945*XC))/DF[4
19080 XQ= XC + XQ
19090 IF DF>3 RETURN
19100 DELTA=1
19110 TX=XQ
19120 GOSUB 15000
19130 IF ABS(PR-FOX) < .00001 THEN XQ=TX : RETURN
19140 IF PR < FOX THEN TX=TX-DELTA
19150 IF PR < FOX THEN DELTA=DELTA/10
19160 IF PR < FOX THEN GOTO 19120
19170 TX=TX+DELTA
19180 GOTO 19120
```

INVERSE - F

```
20000 REM *** I N V E R S E  F - D I S T R I B U T I O N ***
20010 IF DN < 20 OR DD < 20 GOTO 20090
20020 MO=0 : VR=1 : I2=1
20030 GOSUB 17000
20040 H = 2 * ( (1/(DD-1) + 1/(DN-1))[-1 )
20050 LM = (XQ[2 - 3)/6
20060 WX=(XQ * SQR(H + LM))/H - ( (1/(DN-1) - 1/(DD-1) ) * ( LM + 5/6 - 2/(3*H) ))
20070 XQ = EXP(2*WX)
20080 RETURN
20090 IF DN<>1 AND DD<>1 OR DN=1 AND DD=1 GOTO 20190
20100 PH=PR : IF DN=1 THEN PR=(1+PR)/2 ELSE PR=1-PR/2
20110 IF DN=1 THEN DF=DD ELSE DF=DN
20120 GOSUB 19000
20130 XQ=XQ*XQ : IF DD=1 THEN XQ=1/XQ
20140 PR=PH
20150 IF PR>=.99 THEN XQ=.994*XQ : GOTO 20180
20160 IF PR>=.975 AND DN>3 THEN XQ=.995*XQ : GOTO 20180
```

```
20170 IF PR>=.95 AND DN>3 THEN XQ=.996*XQ
20180 RETURN
20190 IF PR>.999 AND DD=1 THEN XQ=9E+36 : RETURN
20200 IF PR>.997 AND DD=1 THEN DELTA=500000 : GOTO 20270
20210 IF PR>.995 AND DD=1 THEN DELTA=15000 : GOTO 20270
20220 IF PR>.99 AND DD=1 THEN DELTA=5000 : GOTO 20270
20230 IF PR>.975 AND DD=1 OR PR>.999 AND DD=2 THEN DELTA=1000 : GOTO 20270
20240 IF PR>.9 AND DD=1 OR PR>.99 AND DD<=2 OR PR>=.999 AND DD<=3 THEN DELTA =100 : GOTO 20270
20250 IF PR>=.99 AND DD<=3 THEN DELTA=50 : GOTO 20270
20260 IF PR>=.975 AND DD<=2 THEN DELTA=50 ELSE DELTA=10
20270 LL=0
20280 FX=DELTA
20290 GOSUB 16000
20300 IF ABS(PR-FOX)<=.00001 THEN XQ=FX : RETURN
20310 IF PR<FOX THEN DELTA=DELTA/2 : FX=DELTA : GOTO 20290
20320 LL=FX : FX=FX+DELTA : UL=FX : GOSUB 16000
20330 IF ABS(PR-FOX)<=.00001 THEN XQ=FX : RETURN
20340 DELTA=(UL-LL)/2
20350 FX=LL+DELTA
20360 GOSUB 16000
20370 IF ABS(PR-FOX)<=.00001 THEN XQ=FX : RETURN
20380 IF PR>FOX THEN LL=FX ELSE UL=FX
20390 GOTO 20340
```

PROGRAM SELECTION MENU

```
22000 REM *** P R O G R A M   S E L E C T I O N   M E N U ***
22010 CLS
22020 PRINT TAB(20)"PROGRAM SELECTION MENU"
22030 PRINT @128,"#1  Data I/O, Transformations, Prob. & Inv. Prob. Distributions":PRINT
22040 PRINT"#2  Sample Stat's, 7-Num-Sum, Plots":PRINT
22050 PRINT"#3  Parametric Tests of Hypotheses and Confidence Intervals":PRINT
22060 PRINT"#4  Robust Estimates and Confidence Intervals":PRINT
22070 PRINT"#5  Multiple Linear Regression":PRINT
22080 PRINT"#6  Analysis of Variance"
22090 JP=896 : KS=6 : GOSUB 760
22100 IF IO=3 GOTO 10
22110 IF IO=4 THEN ZT="2":GOTO 22130
22120 IF IO=1 OR IO=2 THEN ZT="1" ELSE ZT="3"
22130 IF IO=1 OR IO=5 THEN IO=1 ELSE IO=2
22140 CLS
22150 PRINT"INSERT TAPE #";ZT;" INTO ESF-80 AND PRESS SPACE BAR"
22160 IF INKEY$<>" " GOTO 22160
22170 @LOAD IO
```

END OF PROGRAM #3

PROGRAM #4

MENU

```
10 LB$=CHR$(123) : RB$=CHR$(125) : BL$=CHR$(250)
20 REM  *** M A I N   M E N U ***
30 IF I4=1 GOTO 130
40 CLS  : PRINT TAB(20) "***  M E N U  ***" : PRINT
50 PRINT"(1) Robust-t Confidence Intervals for the Population Mean" : PRINT
60 PRINT"(2) Confidence Interval for the Median" : PRINT
70 PRINT"(3) Jackknife (Point Estimates & Confidence Intervals)" : PRINT
80 PRINT"(4) Return to Program Selection Menu
90 KP=768 : KS=4
100 GOSUB 180
110 ON IO GOSUB 500, 2000, 3000, 10000
120 GOTO 20
130 CLS
140 PRINT"None of the subroutines in Program #4 are applicable since only sample statistics were entered.  Pre
ss the SPACE BAR to access  the program selection menu.
150 IF INKEY$<>" " GOTO 150
160 CLS
170 GOTO 10000
180 PRINT @KP+5,"OPTION DESIRED ";LB$;CHR$(176);RB$;
190 ZI=INKEY$ : IF ZI="" GOTO 190ELSE IO=VAL(ZI)
200 PRINT @KP+21,ZI : FOR I=1 TO 100 : NEXT
210 IF IO<1 OR IO>KS THEN PRINT @KP, BL$ : PRINT @KP,"*** NOT A VALID OPTION ***" : FOR I=1 TO 1000 :
 NEXT : PRINT @KP, BL$ : GOTO 180
220 RETURN
```

ROBUST - t

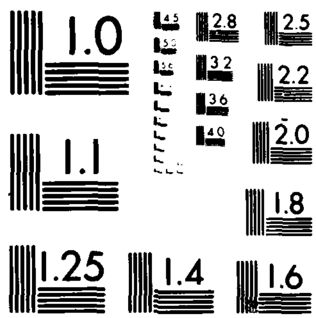
```
500 REM  * * *  R O B U S T - t   S U B R O U T I N E S  * * *
510 CLS
520 PRINT TAB(22)"ROBUST-t TECHNIQUES" : PRINT
530 PRINT"     The two robust techniques listed below are useful in"
540 PRINT"calculating confidence intervals for the population mean"
550 PRINT"when known or suspected outliers violate the assumption of"
560 PRINT"near normality:"
570 PRINT
580 PRINT"     (1) Winsorized-t Technique
590 PRINT"     (2) Trimmed-t Technique
600 KP=896 : KS=2 : GOSUB 180
610 CLS
620 IF IO=1 ZR="Winsorizing" ELSE ZR="Trimming"
630 PRINT"Enter Desired Level of ";ZR;" (10 max)";: INPUT JG
640 IF JG=0 THEN WM=XB : WS=SQR(V1) : GOTO 810
650 FOR I=1 TO JG
660 W(I)=Y(JG+1)
670 WW(I)=Y(IN-JG)
680 NEXT I
690 SM=0
700 S2=0
710 FOR I=1 TO JG
720 SM=SM+W(I)+WW(I)
```

AD-A109 602 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 9/2
STATISTICAL ANALYSIS SOFTWARE FOR THE TRS-80 MICROCOMPUTER.(U)
SEP 81 R P ISBELL

UNCLASSIFIED NL

2 of 2

END
DATE
FILMED
-82
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

```
730 S2=S2+W(I)[2 + WW(I)[2
740 NEXT I
750 FOR I=JG+1 TO IN-JG
760 SM=SM+Y(I)
770 S2=S2+Y(I)[2
780 NEXT I
790 WM=SM/IN
800 WS=SQR((S2-(SM[2)/IN)/(IN-1))
810 ON IO GOSUB 1000, 1500
820 RETURN
```

WINSORIZED - t

```
1000 REM  * * *  W I N S O R I Z E D - t  * * *
1010 CLS
1020 PRINT TAB(25)"WINSORIZED - t"
1030 PRINT @128,"At Level =";JG
1040 PRINT"Winsorized Point Estimate for the Mean =";WM
1050 PRINT : INPUT"Enter Desired Confidence Level (e.g. .95)";AL
1060 IF AL>=1 THEN AL=AL/100
1070 PR=(1+AL)/2
1080 DF=IN-2*JG-1
1090 GOSUB 8000
1100 PRINT : PRINT AL*100;"% CONFIDENCE INTERVAL:"
1110 PRINT : PRINT TAB(15) WM - (IN-1)/(DF) * XQ * WS/SQR(IN);" < True Mean <"; WM + (IN-1)/(DF) * XQ * W
S/SQR(IN)
1120 PRINT @960,"PRESS SPACE BAR TO RETURN TO MAIN MENU";
1130 IF INKEY$ <> " " GOTO 1130
1140 RETURN
```

TRIMMED - t

```
1500 REM  * * *  T R I M M E D - t  * * *
1510 CLS : PRINT TAB(25)"TRIMMED-t"
1520 PRINT @128,"At Level =";JG
1530 PRINT"Trimmed Point Estimate for the Mean =";
1540 SM=0
1550 FOR I=JG+1 TO IN-JG
1560 SM=SM+Y(I)
1570 NEXT I
1580 TM=SM/(IN-2*JG)
1590 RW=(((IN-1)/(IN-2*JG-1))*WS[2)
1600 PRINT TM
1610 PRINT : INPUT"Enter Desired Confidence Level (e.g. .95)";AL
1620 IF AL>=1 THEN AL=AL/100
1630 DF=IN-2*JG-1
1640 PR=(1+AL)/2
1650 GOSUB 8000
1660 PRINT : PRINT TAB(15) TM - XQ*SQR(RW/(IN-2*JG));"< True Mean <";TM + XQ*SQR(RW/(IN-2*JG))
1670 PRINT @960,"PRESS SPACE BAR TO RETURN TO MAIN MENU";
1680 IF INKEY$ <> " " GOTO 1680
1690 RETURN
```

CONFIDENCE INTERVAL FOR THE MEDIAN

```
2000 REM  * * *  C O N F I D E N C E   I N T E R V A L S   F O R   T H E   M E D I A N  * * *
2010 CLS
2020 PRINT @15,"CONFIDENCE INTERVAL FOR THE MEDIAN"
2030 PRINT @128,"    Calculation  of  a  confidence interval for the median is"
2040 PRINT"based on the binomial distribution which is, of course, a dis-"
```

```
2050 PRINT"crete distribution.  An interval will be calculated such that"
2060 PRINT"the confidence level you specify will be met or exceeded."
2070 PRINT : INPUT"Enter Desired Confidence Level (e.g. .95)";AL
2080 IF AL>=1 THEN AL=AL/100
2090 CLS : PRINT @471,"CALCULATING"
2100 P=.5
2110 PR=0
2120 IF IN<=30 BB=0 : GOTO 2180
2130 IF IN>=150 BB=55 : GOTO 2170
2140 IF IN>=100 BB=35 : GOTO 2170
2150 IF IN>=60 BB=20 : GOTO 2170
2160 IF IN>=40 THEN BB=12 ELSE BB=8
2170 MU=.5 : VAR=.25/IN : GOTO 2210
2180 GOSUB 2500
2190 PH = PR
2200 PR=PR + BC * P[IN : GOTO 2220
2210 PH=PR : NX=(BB+.5)/IN : GOSUB 5000: PR=FOX
2220 IF PR < (1 - AL)/2 THEN BB=BB+1 : IF IN>30 GOTO 2210ELSE GOTO 2180
2230 PR=1-2*PH
2240 CLS
2250 IF PR=1 THEN PR=.995
2260 PRINT @10,"";:PRINT USING"###.#";PR*100;:PRINT"% CONFIDENCE INTERVAL FOR THE MEDIAN"
2270 IF BB=0 THEN BB=1
2280 PRINT @148, Y(BB);"< Median <"; Y(IN-BB+1)
2290 FOR BB = R1 TO S1-1
2300 PRINT @960,"PRESS SPACE BAR TO RETURN TO MENU";
2310 IF INKEY$ <> " " GOTO 2310
2320 RETURN
2500 REM *** B I N O M I A L   C O E F F I C I E N T ***
2510 BC=0 : B=0 : SB=BB : QB=IN-BB
2520 FOR BI=1 TO IN
2530 IF SB>=QB GOTO 2590
2540 IF SB>=BI THEN B=B+LOG(BI)
2550 IF QB>BI GOTO 2630
2560 IF QB=BI THEN BC=-B : GOTO 2630
2570 BC=BC + LOG(BI)
2580 GOTO 2630
2590 IF QB>=BI THEN B=B+LOG(BI)
2600 IF SB>BI THEN GOTO 2630
2610 IF SB=BI THEN BC=-B : GOTO 2630
2620 BC=BC+LOG(BI)
2630 NEXT BI
2640 BC = EXP(BC)
2650 RETURN
```

JACKKNIFE ESTIMATES & CONFIDENCE INTERVALS

```
3000 REM *** J A C K K N I F E   S U B R O U T I N E ***
3010 CLS : PRINT TAB(22) "J A C K K N I F E"
3020 PRINT : PRINT"The jackknife technique is useful for reducing bias in the     estimates of parameters  and i
n obtaining robust interval      estimates.
3030 PRINT
3040 PRINT"NOTE 1: Estimate & Conf. Int. for Variance are based on Jack-   knife of log(base 10) of s" : PRINT
3050 PRINT"NOTE 2: Jackknife Est. & Conf. Int. for the Mean are identical  to classical calculations and are incl
uded here for convenience only.
3060 PRINT
3070 PRINT"    ";LB$;"1";RB$;" Mean"
3080 PRINT"    ";LB$;"2";RB$;" Variance"
3090 KP=960 : KS=2 : GOSUB 180
3100 CLS:IF IO=1 THEN PE=XB : IH=1 :  SD=SQR(V1) : GOTO 3220
3120 PM=LOG(SQR(V1))/LOG(10)
3130 PE=0 : S2=0
```

```
3140 FOR I=1 TO IN STEP IH
3150 GOSUB 3450
3160 SV=(IN/IH)*PM - (IN/IH-1)*LO1
3170 S2=S2 + SV*SV
3180 PE=PE + SV
3190 NEXT I
3200 SD=SQR( (S2 - (PE[2)/(IN/IH))/(IN/IH-1) )
3210 PE=PE*IH/IN
3215 CLS
3220 PRINT @0,CHR$(253) : PRINT"Point estimate of ";
3230 IF IO=1 ZP="Mean" ELSE ZP="Variance"
3240 PRINT ZP;" =";
3250 IF IO=1 PRINT PE ELSE PRINT 10[(PE+PE)
3260 IF IO=2 PRINT"Point estimate of Std. Deviation =";10[PE
3270 PRINT : INPUT"Enter the desired Confidence Level  (e.g. .95)";AL
3280 IF AL>=1 THEN AL=AL/100
3290 PR=1 - (1-AL)/2 : IF PR>1 PR=PR/100
3300 DF=IN/IH-1 : I2=1 : GOSUB 8000
3310 L=PE - XQ * SD/SQR(IN/IH) : IF IO=2 L=10[(L+L)
3320 U=PE + XQ * SD/SQR(IN/IH) : IF IO=2 U=10[(U+U)
3330 PRINT : PRINT AL*100;"% Confidence Interval:"
3340 PRINT"     ";L;" < True Value of ";ZP;" <";U
3350 IF IO=2 PRINT"     ";SQR(L);" < True Value of Std. Deviation <";SQR(U)
3360 PRINT : PRINT : PRINT : PRINT"*** PRESS SPACE BAR TO RETURN TO MAIN MENU"
3370 IF INKEY$ <> " " GOTO 3370
3380 RETURN
3400 SP=0 : FOR I=1 TO 5
3405 IF IN/I=INT(IN/I) THEN PG(I)=1 ELSE PG(I)=0
3410 IF PG(I)=1 THEN SP=SP+1
3415 NEXT I
3420 CLS : IF SP>1 THEN 3432 ELSE PRINT "    Since";IN;"(the sample size) is a prime number it is only"
3425 PRINT"possible to jackknife in groups of 1. For large sample sizes, computation time is excessive.";
3430 PRINT:GOTO 3433
3432 PRINT"     You may jackknife in groups as listed below. Computation  time is a function of both the sample
size and group size and   becomes excessive with large samples and small groups."
3433 PRINT:PRINT"e.g.    Sample Size   Group Size   Approx. Time":PRINT"          100        1         25 min.":IF
SP>1 PRINT"          100        2          8 min.":PRINT"           50        1          4 min."
3434 PRINT:PRINT"You may:  ";LB$;"Q";RB$;"uit"
3435 JL=1:PRINT"         or":PRINT"Jackknife in groups of:"
3436 FOR I=1 TO 5
3438 IF PG(JL)=1 THEN PRINT"         ";LB$;JL;RB$; ELSE JL=JL+1 : IF JL<=5 THEN 3438
3442 IF PG(JL)=1 THEN JL=JL+1: NEXT I
3444 PRINT @965,"Indicate Option ";LB$;CHR$(176);RB$;
3446 O$=INKEY$ : IF O$="" OR O$=CHR$(13) THEN 3446 ELSE PRINT @ 982,O$;:FOR I=1 TO 100:NEXT
3447 IF O$="Q" OR O$="q" THEN 40 ELSE FOR IH=1 TO 5
3448 IF VAL(O$)=IH AND PG(IH)=1 THEN RETURN
3449 NEXT IH : GOTO 3444
3450 LO1=0 : SM=0
3460 FOR J=1 TO IN
3470 IF J<I OR J>=I+IH THEN SM=SM+RY(J) : LO1=LO1+RY(J)[2
3480 NEXT J
3490 LO1=(LO1 - (SM[2)/(IN-IH))/(IN-IH-1)
3500 LO1=LOG(SQR(LO1))/LOG(10)
3510 RETURN
3550 FOR I=1 TO IN : RY(I)=Y(I) : NEXT : IF IH=1 RETURN
3555 FOR I=IN TO 1 STEP -1
3560 IU=RND(IN)
3563 RT=RY(I) : RY(I)=RY(IU) : RY(IU)=RT
3570 NEXT I
3575 RETURN
```

NORMAL DISTRIBUTION

```
5000 REM *** N O R M A L  D I S T R I B U T I O N ***
5010 SD=SQR(VAR)
5020 XN=(NX-MU)/SD:AX=ABS(XN)
5030 XZ=.3989423*EXP((-AX*AX)/2)
5040 NT=1/(1+.2316419*AX)
5050 QOX=XZ*((.3193815*NT)-(.3565638*NT[2)+(1.781478*NT[3)-(1.821256*NT[4)+(1.330274*NT[5))
5060 IF XN<0 GOTO 5090
5070 FOX=1-QOX
5080 RETURN
5090 FOX=QOX
5100 QOX=1-FOX
5110 RETURN
```

t - DISTRIBUTION

```
6000 REM *** t - D I S T R I B U T I O N ***
6010 IF TX<0 I3=1 ELSE I3=0
6020 IF DF>25 GOTO 6450
6030 THETA=ATN(ABS(TX)/SQR(DF))
6040 IF DF<>1 GOTO 6090
6050 AT=.6366198*THETA
6060 FOX=(AT+1)/2
6070 QOX=1-FOX
6080 GOTO 6400
6090 IF INT(DF/2)=DF/2 GOTO 6220
6100 IF DF=3 GOTO 6370
6110 TS=0 : TP=1 : TC=COS(THETA)
6120 FOR I=1 TO INT((DF-3)/2)
6130 TP=TP*2*I/(1+2*I)
6140 TC=TC*(COS(THETA)[2)
6150 TS=TS+(TP*TC)
6160 NEXT I
6170 TS=TS+COS(THETA)
6180 AT=.6366198*(THETA+(SIN(THETA)*TS))
6190 FOX=(AT+1)/2
6200 QOX=1-FOX
6210 GOTO 6400
6220 IF DF<>2 GOTO 6270
6230 AT=SIN(THETA)
6240 FOX=(AT+1)/2
6250 QOX=1-FOX
6260 GOTO 6400
6270 TS=0 : TP=1 : TC=1
6280 FOR I=1 TO INT((DF-2)/2)
6290 TP=TP*((2*I)-1)/(2*I)
6300 TC=TC*(COS(THETA)[2)
6310 TS=TS+(TP*TC)
6320 NEXT I
6330 AT=SIN(THETA)*(1+TS)
6340 FOX=(AT+1)/2
6350 QOX=1-FOX
6360 GOTO 6400
6370 AT=.6366198*(THETA+(SIN(THETA)*COS(THETA)))
6380 FOX=(AT+1)/2
6390 QOX=1-FOX
6400 FX=TX
6410 IF I3=0 RETURN
6420 QOX=FOX
6430 FOX=1-QOX
6440 RETURN
6450 NX=( TX * (1 - .25/DF) )/SQR(1 + .5*TX[2/DF)
```

```
6460 MU=0 : VAR=1 : I2=1 : GOSUB 5000
6470 PX=TX
6480 RETURN
```

INVERSE NORMAL

```
7000 REM *** I N V E R S E  N O R M A L  D I S T R I B U T I O N ***
7010 IF PR>.5 THEN RL=SQR(LOG(1/((1-PR)[2))) ELSE RL=SQR(LOG( 1/(PR[2)))
7020 XQ=RL-((2.515517+(.802853*RL)+(.010328*RL[2))/(1+(1.432788*RL)+(.189269*RL[2)+(.001308*RL[3)))
7030 IF PR>=.5 THEN XQ=MO+(SQR(VR)*XQ) ELSE XQ=MO-(SQR(VR)*XQ)
7040 RETURN
```

INVERSE - t

```
8000 REM *** I N V E R S E  t - D I S T R I B U T I O N ***
8010 MO=0 : VR=1 : I2=1
8020 GOSUB 7000
8030 XC=XQ
8040 XQ=(.25 * (XC[3 + XC))/DF
8050 XQ=XQ + ((1/96) * (5*XC[5 + 16*XC[3 + 3*XC))/DF[2
8060 XQ=XQ + ((1/384) * (3*XC[7 + 19*XC[5 + 17*XC[3 -15*XC))/DF[3
8070 XQ=XQ + ((1/92160) * (79*XC[9 + 776*XC[7 + 1482*XC[5 -1920*XC[3 -945*XC))/DF[4
8080 XQ= XC + XQ
8090 IF DF>3 RETURN
8100 DELTA=1
8110 TX=XQ
8120 GOSUB 6000
8130 IF ABS(PR-FOX) < .00001 THEN XQ=TX : RETURN
8140 IF PR < FOX THEN TX=TX-DELTA
8150 IF PR < FOX THEN DELTA=DELTA/10
8160 IF PR < FOX THEN GOTO 8120
8170 TX=TX+DELTA
8180 GOTO 8120
```

PROGRAM SELECTION MENU

```
10000 REM *** P R O G R A M  S E L E C T I O N  M E N U ***
10010 CLS:PRINT TAB(20)"PROGRAM SELECTION MENU"
10020 PRINT @128,"#1  Data I/O, Transformations, Prob. & Inv. Prob. Distributions":PRINT
10030 PRINT"#2  Sample Stat's, 7-Num-Sum, Plots":PRINT
10040 PRINT"#3  Parametric Tests of Hypotheses and Confidence Intervals":PRINT
10050 PRINT"#4  Robust Estimates and Confidence Intervals": PRINT
10060 PRINT"#5  Multiple Linear Regression":PRINT
10070 PRINT"#6  Analysis of Variance"
10080 KP=896 : KS=6 : GOSUB 180
10090 IF IO=4 GOTO 20
10100 IF IO=3 THEN ZT="2" : GOTO 10120
10110 IF IO=1 OR IO=2 THEN ZT="1" ELSE ZT="3"
10120 CLS
10130 IF IO=1 OR IO=3 OR IO=5 THEN IO=1 ELSE IO=2
10140 CLS
10150 PRINT"INSERT TAPE #";ZT;" INTO ESF-80 AND PRESS SPACE BAR"
10160 IF INKEY$<>" " GOTO 10160
10170 @LOAD1
```

END OF PROGRAM #4

PROGRAM #5

MENU

```
5 CLEAR
10 DIM MM(100,10), VC(100,10), VTV(10,100), RV(100), RH(100), E(100), H(100), ID(20)
20 DEFSTR Z : DEFINT I-K
25 LB$=CHR$(123) : RB$=CHR$(125) : BL$=CHR$(252)
30 REM *** P R O G R A M  D I S C R E P T I O N ***
40 CLS
50 PRINT"   * M U L T I P L E  L I N E A R  R E G R E S S I O N *"
60 PRINT : PRINT "    This program fits a linear model to the data by the method of Ordinary Least Squares (OLS ). Parameters and statistics cal-culated include the equation coefficients, standardized coeffi- cients, ";
70 PRINT "partial F statistics for the coefficents, correlation  matrix, Hat Matrix (diagonal elements) and ANOV A table.
80 PRINT
90 PRINT"  MODEL:  Y = B0 + B1 * X1 + B2 * X2 + ... + Bn * Xn
100 PRINT : PRINT"  WHERE:  Y =  Response Variable
110 PRINT"          X1 ... Xn = Carrier Variables
120 PRINT"          B0 =  Constant ( Y Intercept )
130 PRINT"          B1 ... Bn = Coefficients of Carrier Variables
140 PRINT : PRINT"* PRESS SPACE BAR TO CONTINUE";
150 IF INKEY$ <> " " GOTO 150
160 CLS
170 FOR I=18 TO 59 : SET(I,3) : NEXT
175 PRINT @10,"PROGRAM LIMITATIONS";
180 PRINT @128,"(a)  No more than 9 carrier variables
190 PRINT @256,"(b)  No more than 100 observations
200 PRINT @465,"MENU:": FOR I=30 TO 46: SET(I,24):NEXT
210 PRINT @576,"(1)  Data I/O
220 PRINT : PRINT"(2)  Program Selection Menu"
230 KP=896 : KS=2 : GOSUB 300
240 ON IO GOTO 400, 10000
```

OPTION SELECTION

```
300 REM *** O P T I O N  S E L E C T I O N
310 PRINT @KP+5,"OPTION DESIRED ";LB$;CHR$(176);RB$;
320 ZI=INKEY$ : IF ZI="" GOTO 320ELSE IO=VAL(ZI)
330 PRINT @KP+21, ZI; : FOR I=1 TO 100 : NEXT
340 IF IO<1 OR IO>KS THEN PRINT @KP, BL$; : PRINT @KP,"*** NOT A VALID OPTION ***";: FOR I=1 TO 1000 : NEXT : PRINT @KP, BL$;: GOTO 310
350 RETURN
360 PRINT @KP+5,LB$;" ";L$;" or ";R$;RB$;"  ";LB$;CHR$(176);RB$;
370 ZQ=INKEY$ : IF ZQ="" GOTO 370ELSE PRINT @KP+17, ZQ; : FOR I=1 TO 100 : NEXT
380 IF ZQ<>L$ AND ZQ<>CHR$(ASC(L$)-32) AND ZQ<>R$ AND ZQ<>CHR$(ASC(R$)-32) THEN 360
390 RETURN

400 REM *** I N P U T / O U T P U T  O P T I O N S ***
410 CLS
420 PRINT TAB(20)"INPUT/OUTPUT OPTIONS"
430 PRINT @128,"(1) Input Data via Keyboard
440 PRINT"(2) Input Data via Tape
450 PRINT"(3) Save Data to Tape
460 KP=896 : KS=3 : GOSUB 300
470 ON IO GOTO 1000, 500, 700
```

INPUT DATA VIA TAPE

```
500 REM *** I N P U T  D A T A  V I A  T A P E ***
510 CLS : PRINT"Insert DATA TAPE into ESF-80 and press SPACE BAR"
520 IF INKEY$<>" " GOTO 520
530 PRINT @132,"Data File Number ";LB$;CHR$(176);RB$;
540 ZI=INKEY$ : IF ZI="" THEN 540: ELSE FI=VAL(ZI)
550 PRINT @150, ZI;: FOR I=1 TO 100 : NEXT
560 IF FI<1 THEN 530
570 @CLOSE
580 @OPEN FI
590 @INPUT CV,IN,YM,Y2
600 FOR I=1 TO IN
610 @INPUT RV(I)
620 FOR J=1 TO CV
625 IF I<>1 THEN 630
628 @INPUT MU(J),X2(J)
630 @INPUT MM(I,J)
640 NEXT J
650 NEXT I
660 @CLOSE
665 TA$="Y"
670 CLS : PRINT"CALCULATING CORRELATION MATRIX..." : GOTO1230
```

SAVE DATA TO TAPE

```
700 REM *** S A V E  D A T A  T O  T A P E ***
710 CLS : PRINT"Insert DATA TAPE into ESF-80 and press SPACE BAR"
720 IF INKEY$<>" " GOTO 720
730 PRINT @132,"File Number to Save Data Under ";LB$;CHR$(176);RB$;
740 ZI=INKEY$ : IF ZI="" THEN 740ELSE FI=VAL(ZI)
750 PRINT @164, ZI;: FOR I=1 TO 100 : NEXT
760 IF FI<1 THEN 730
770 @CLOSE
780 @OPEN FI
790 @PRINT CV,IN,YM,Y2
800 FOR I=1 TO IN
810 @PRINT RV(I)
820 FOR J=1 TO CV
825 IF I<>1 THEN 830
828 @PRINT MU(J),X2(J)
830 @PRINT MM(I,J)
840 NEXT J
850 NEXT I
860 @CLOSE
870 PRINT @512,"OPTIONS:"
880 @CLOSE
890 PRINT"   (1) Continue with Regression
900 PRINT"   (2) Return to Program Menu
910 PRINT"   (3) Go to Program Selection Menu
920 KP=896 : KS=3 : GOSUB 300
930 ON IO GOTO 1500, 160, 10000
```

INPUT DATA VIA KEYBOARD

```
1000 REM *** I N P U T  D A T A  V I A  K E Y B O A R D ***
1010 CLS
1020 INPUT"Number of Carrier Variables to be entered";CV
1030 IF CV>9 PRINT"*** ERROR ***  NO MORE THAN 9 CARRIERS ALLOWED":GOTO 1020
1040 IF CV<1 PRINT"*** ERROR ***  MUST HAVE AT LEAST 1 CARRIER VARIABLE":GOTO 1020
1045 FOR I=1 TO CV : MU(I)=0 : XS(I)=0 : X2(I)=0 : SX(I)=0 : NEXT I
1050 INPUT"Enter the Number of Observations";IN
1060 IF IN<=CV+1 THEN PRINT"*** ERROR ***  MIN. NUMBER OF OBSERVATIONS =";CV+2 : GOTO 1050
```

```
1070 CLS
1085 YS=0 : Y2=0 : SY=0 : YM=0
1090 FOR I=1 TO IN
1100 PRINT"OBSERVATION #";I
1120 PRINT"Enter the observed value for the response variable, Y";:INPUT RV(I)
1122 YM=YM+RV(I) : SY=RV(I)-YS : YS=YS+SY/I : Y2=Y2+SY*(RV(I)-YS)
1130 FOR J=1 TO CV
1140 PRINT"Enter the Value for carrier variable, X";MID$(STR$(J),2,2);:INPUT MM(I,J)
1142 MU(J)=MU(J)+MM(I,J) : SX(J)=MM(I,J)-XS(J) : XS(J)=XS(J)+SX(J)/I : X2(J)=X2(J)+SX(J)*(MM(I,J)-XS(J))
1150 NEXT J
1160 PRINT
1170 NEXT I
```

__CALCULATE AND DISPLAY CORRELATION MATRIX__

```
1180 CLS : PRINT"CALCULATING CORRELATION MATRIX..."
1190 YM=YM/IN
1200 FOR J=1 TO CV
1210 MU(J)=MU(J)/IN
1220 NEXT J
1225 TA$="N"
1230 FOR I=1 TO IN
1240 IF TA$<>"Y" THEN RV(I)=(RV(I)-YM)/SQR(Y2)
1250 FOR J=1 TO CV
1260 VC(I,J)=(MM(I,J)-MU(J))/SQR(X2(J))
1280 NEXT J
1290 NEXT I
1295 FOR I=1 TO IN : VC(I,J)=RV(I) : NEXT I
1300 IV=CV+1 : RET$="Y" : GOSUB 2000
1310 RET$="N" : EPS=1E-5
1320 CLS : PRINT"CORRELATION MATRIX:" : PRINT
1325 IF CV<=5 THEN PRINT"      "; ELSE PRINT"    ";
1330 FOR I=1 TO CV : PRINT "   X";RIGHT$(STR$(I),1); : IF CV<=5 THEN PRINT "     "; ELSE PRINT" ";
1335 NEXT : PRINT"  Y"
1340 FOR I=1 TO IV
1350 FOR J=1 TO IV
1355 IF CV>5 THEN P$="#.###" ELSE P$="#.####"
1360 IF J=1 AND I<=CV THEN PRINT "X";RIGHT$(STR$(I),1);"    "; ELSE IF J=1 PRINT "Y     ";
1362 IF ABS(VV(I,J)-1)<EPS AND CV>5 THEN PRINT "  1   "; : GOTO 1380
1364 IF ABS(VV(I,J)-1)<EPS THEN PRINT"  1     "; : GOTO 1380
1370 PRINT USING P$;VV(I,J); : IF CV<=5 THEN PRINT"    "; ELSE PRINT" ";
1380 NEXT J
1390 PRINT
1400 NEXT I
1410 PRINT @969,"Press SPACE BAR to Continue";
1420 IF INKEY$<>" " THEN 1420
```

__SELECT SUBSET OF CARRIER VARIABLES__

```
1500 REM *** S E L E C T  S U B S E T  O F  C A R R I E R S ***
1510 IF CV=1 THEN 1580
1520 CLS : PRINT"  The data array entered contains";CV;"carrier variables. You"
1530 PRINT"may regress on all";CV;"carriers or any subset."
1540 PRINT @197,LB$;"A";RB$;"ll  or  ";LB$;"S";RB$;"ubset  ";LB$;CHR$(176);RB$;
1550 HC$=INKEY$ : IF HC$="" THEN GOTO 1550ELSE PRINT @221,HC$ : FOR I=1 TO 100 : NEXT
1560 IF HC$<>"A" AND HC$<>"a" AND HC$<>"S" AND HC$<>"s" THEN 1540
1570 IF HC$="S" OR HC$="s" THEN 1600
1580 FOR I=1 TO CV : U$(I)="Y" : NEXT : GOTO 1810
1600 REM ***  CHOOSE SUBSET OF CARRIERS
1620 CLS : IF IN>=5 THEN JD=5 ELSE JD=IN
1630 PRINT"To facilitate recognition, the first";JD;"values for each carrier" : PRINT"are displayed below.  When
prompted indicate whether you want"
1640 PRINT"to  ";LB$;"R";RB$;"etain  or  ";LB$;"D";RB$;"elete  each carrier for the regression."
1650 KP=256
```

```
1660 FOR I=1 TO CV
1670 PRINT @KP+5,"X";MID$(STR$(I),2,2);"  ";
1680 FOR J=1 TO JD
1690 PRINT MM(J,I);:IF J<>JD PRINT ",";
1700 NEXT J
1710 KP=KP+64
1720 NEXT I
1730 KP=256
1740 FOR I=1 TO CV
1750 PRINT @KP,LB$;CHR$(176);RB$;
1760 RT$=INKEY$ : IF RT$="" GOTO 1760ELSE PRINT @KP+1,RT$;
1770 IF RT$<>"R" AND RT$<>"r" AND RT$<>"D" AND RT$<>"d" THEN 1750
1780 IF RT$="R" OR RT$="r" THEN U$(I)="Y" ELSE U$(I)="N"
1790 KP=KP+64
1800 NEXT I
1810 U$(0)="Y"
1820 IV=0
1830 FOR I=1 TO CV
1840 IF U$(I)="Y" THEN IV=IV+1 : IW(IV)=I
1850 NEXT I
1860 J=0
1870 PRINT @896,"Regress on: ";
1880 FOR I=1 TO CV
1890 IF U$(I)="Y" THEN IP(J+1)=I : J=J+1
1900 NEXT I
1910 IF IV=0 THEN PRINT @896,"*** ERROR ***  YOU HAVE DELETED ALL CARRIERS":FOR I=1 TO 2500 : NEXT
 : GOTO 1500
1920 FOR I=1 TO IN
1930 FOR J=1 TO IV
1940 IF J=IV AND I<=IV THEN PRINT "X";MID$(STR$(IW(I)),2,2);", ";
1950 VC(I,J)=(MM(I,IP(J))-MU(IP(J)))/SQR(X2(IP(J)))
1960 NEXT J
1970 NEXT I
1980 PRINT @960,"Calculating...";
```

CALCULATION OF OLS ESTIMATES

CALCULATE X' X

```
2000 REM *** C A L C U L A T I O N  O F  L E A S T  S Q U A R E S  E S T I M A T O R ***
2010 REM  BETA.hat = ( X' X ).inverse  X' Y
2030 REM  *** C A L C U L A T E  X'X ***
2040 FOR I=1 TO IV
2050 FOR J=1 TO IV
2060 VV(I,J)=0 : VI(I,J)=0
2070 NEXT J
2080 NEXT I
2090 FOR I=1 TO IV
2100 FOR J=1 TO IV
2110 SUM=0
2120 FOR IK=1 TO IN
2130 SUM=SUM + VC(IK,I)*VC(IK,J)
2140 NEXT IK
2150 VV(I,J)=SUM
2160 NEXT J
2170 NEXT I
2180 IF RET$="Y" THEN RETURN
```

CALCULATE (X'X)inverse

```
2500 REM  *** C A L C U L A T E  ( X' X )inverse  ***
2510 FOR I=1 TO IV : VI(I,I)=1 : NEXT
```

```
3240 SUM=SUM+VTV(I,IK)*RV(IK)
3250 NEXT IK
3260 B(I) = SUM
3270 NEXT I
```

TRANSFORM BETA COEFFICIENTS TO REGRESSION EQUATION COEFFICIENTS

```
3300 REM  * * *  TRANSFORM BETA COEFFICIENTS TO REGRESSION EQUATION COEFFICIENTS  * * *
3305 B(0)=YM : SC(0)=YM
3310 FOR I=1 TO IV
3320 SC(I)=B(I)
3330 B(I)=SC(I)*SQR(Y2)/SQR(X2(IP(I)))
3340 B(0)=B(0)-B(I)*MU(IP(I))
3350 NEXT I
```

CALCULATE ELEMENTS OF ANOVA TABLE

```
3500 REM *** C A L C U L A T E  E L E M E N T S  O F  A N O V A  T A B L E ***
3510 RSS=0 : ESS=0 : E2S=0
3520 TSS=Y2
3570 FOR I=1 TO IN
3580 SUM=0 : S2M=0
3590 FOR IK=1 TO IV
3600 SUM=SUM+MM(I,IP(IK))*B(IK)
3605 S2M=S2M+VC(I,IK)*SC(IK)
3610 NEXT IK
3620 SUM=SUM+B(0)
3630 RH(I)=SUM
3640 ESS=ESS+(RV(I)*SQR(Y2)+YM-SUM)*(RV(I)*SQR(Y2)+YM-SUM)
3645 E2S=E2S+(RV(I)-S2M)*(RV(I)-S2M)
3650 E(I)=(RV(I)*SQR(Y2)+YM)-SUM
3660 NEXT I
3665 RSS=TSS-ESS
3670 MRS=RSS/(IV)
3680 MES=ESS/(IN-IV-1)
3685 M2S=E2S/(IN-IV-1)
3690 F=MRS/MES
3700 R2=RSS/TSS
3710 FOR I=1 TO IV
3720 T(I)=SC(I)/(SQR(M2S * VI(I,I)))
3770 SE(I)=SQR(M2S * VI(I,I)) * SQR(Y2)/SQR(X2(IP(I)))
3780 NEXT I
```

DISPLAY ANOVA TABLE, REGRESSION EQUATION, OTHER STATISTICS

```
4000 REM *** P R I N T  O L S  S O L U T I O N ,  A N O V A ,  &  O T H E R  S T A T I S T I C S ***
4010 CLS
4020 PRINT @26,"A N O V A"
4030 PRINT" Source    DF    Sum of Squares    Mean Square   F-Ratio
4040 FOR I=5 TO 15: SET(I,6): NEXT: FOR I=26 TO 29: SET(I,6): NEXT: FOR I=40 TO 67: SET(I,6): NEXT: FOR I=78
 TO 99: SET(I,6): NEXT: FOR I=106 TO 120: SET(I,6): NEXT
4050 PRINT @192,"Regression";:PRINT @204,IV;:PRINT @213,RSS;: PRINT @231,MRS;:PRINT @243," ";:IF F>999.99
9 THEN PRINT USING"##.#####[[[[";F ELSE PRINT USING"####.###";F
4060 PRINT @258,"Error";:PRINT @268,IN-(IV+1);:PRINT @277,ESS;:PRINT@295,MES
4070 PRINT @322,"Total";:PRINT @332,IN-1;:PRINT @341,TSS
4080 PRINT : PRINT "The Regression Equation is:"
4090 PRINT"  Y = ";
4100 FOR I=0 TO IV
4110 IF I <> 0 GOTO 4140
4120 IF B(0)>=0 THEN PRINT "  "; ELSE PRINT "- ";
4130 GOTO 4150
4140 IF B(I) >= 0 THEN PRINT" + "; ELSE PRINT" - ";
4150 PV=ABS(B(I))
```

```
4160 IF PV>9999.9999 OR PV<.0001 THEN PU$="#.####[[[[" ELSE PU$="#####.####"
4170 PRINT USING PU$;PV;
4180 IF I<>0 PRINT" X";MID$(STR$(IW(I)),2,2);" "; ELSE PRINT "   ";
4190 IF I=2 OR I=5 OR I=8 THEN  PRINT" " : PRINT"    ";
4200 NEXT I
4205 IF IV<=5 THEN PRINT @704,""; ELSE PRINT @768,"";
4210 PRINT "R-Square:     ";:PRINT USING"#.####";R2;:PRINT"      ";"Adjusted R-Square:    ";:PRINT USING"#
.####";1-(1-R2)*(IN-1)/(IN-IV-1)
4220 PRINT"Std. Error:";:PRINT USING"#####.####";SQR(MES)
4225 PRINT"Std. Error as % of Response Mean:";:PRINT USING"#######.###";(SQR(MES)/YM)*100;:PRINT"%";
4230 PRINT @960,"PRESS SPACE BAR TO DISPLAY ADDITIONAL STATISTICS";
4240 IF INKEY$<>" " GOTO 4240
4250 REM * * * PRINT COEFF's, t-STAT's, STD. ERROR of COEFF's, & STANDARDIZED COEFF's * * *
4260 CLS
4270 PRINT"Var.  Coefficient    Partial F    Std. Error    Beta Coeff"
4280 PRINT"----  -----------  -----------  -----------  ------------"
4290 U1$="##########.####" : U2$="#.####[[[["
4300 FOR I=1 TO IV
4310 PRINT"X";MID$(STR$(IW(I)),2,2);
4320 IF ABS(B(I))>999999 OR ABS(B(I))<.0001 THEN PRINT "    ";:PRINT USING U2$;B(I);:GOTO 4340
4330 PRINT USING U1$;B(I);
4340 IF ABS(T(I)*T(I))>999999 OR ABS(T(I)*T(I))<.0001 THEN PRINT"    ";:PRINT USING U2$;T(I)*T(I);: GOTO 43
60
4350 PRINT USING U1$;T(I)*T(I);
4360 IF ABS(SE(I))>999999 OR ABS(SE(I))<.0001 THEN PRINT"    ";:PRINT USING U2$; SE(I);:GOTO 4380
4370 PRINT USING U1$; SE(I);
4380 IF ABS(SC(I))>999999 OR ABS(SC(I))<.0001 THEN PRINT"    ";:PRINT USING U2$; SC(I) : GOTO 4400
4390 PRINT USING U1$; SC(I)
4400 NEXT I
4410 PRINT @896,"Do you want to see the Diagonal Elements of the Hat Matrix?";
4420 L$="y" : R$="n" : KP=960 : GOSUB 360
4430 IF ZQ="N" OR ZQ="n" THEN PRINT @896, BL$;: PRINT @960, BL$;:GOTO 4970
```

DISPLAY DIAGONAL ELEMENTS OF HAT MATRIX

```
4500 REM *** D I S P L A Y  D I A G O N A L  E L E M E N T S  O F  H A T  M A T R I X ***
4510 CLS
4520 PRINT "  The diagonal elements of the Hat Matrix ( H = X(X'X)inv. X' )give a measure of the influence of th
e i th  observation of theresponse variable on the corresponding fitted value.";
4530 PRINT"  A 'large'value of h(i,i) is indicative of an outlier. This, of course, does not imply that the data po
int is necessarily a 'bad' point,but should alert you to inquire into its nature."
4540 PRINT : PRINT"   A reasonable rule of thumb for what is considered a large  H(i,i) is any value larger than
twice the rank of the hat ma-  trix divided by the number of observations."
4550 PRINT @704, ".   i.e.   Consider H(i,i) >";2*IV/IN;"  as large"
4560 IF LH$="N" GOTO 4590
4570 PRINT @834,"NOTE: A blinking cursor will highlight 'large'  H(i,i)'s"
4580 FOR JG=1 TO 10 : PRINT @832,CHR$(143);: FOR JH=1 TO 25 : NEXT JH: PRINT @832," ";: FOR JH=1 TO 25 :
NEXT JH :NEXT JG : PRINT @832,"";
4590 PRINT @960,"Press SPACE BAR to display the H(i,i) 's";
4600 IF INKEY$ <> " " GOTO 4600
4610 CLS
4620 PRINT"Diagonal Elements of Hat Matrix:"
4630 FOR I=1 TO IN STEP 3
4640 IF INT(I/37) <> I/37 AND INT(I/73) <> I/73 GOTO 4700
4650 IF LH$="Y" GOSUB 4810
4660 PRINT @960,"Press SPACE BAR to display remaining elements";
4670 IF INKEY$ <> " " GOTO 4670
4680 CLS
4690 PRINT"Diagonal Elements of Hat Matrix (continued):"
4700 FOR J=0 TO 2
4710 IF I+J > IN GOTO 4790
4720 IF J=0 GOTO 4740
4730 IF J=1 PRINT TAB(21); ELSE PRINT TAB(42);
4740 J$=MID$(STR$(I+J),2,3)
4750 PRINT"H(";J$;",";J$;") =";H(I+J);
```

```
4760 NEXT J
4770 PRINT
4780 NEXT I
4790 IF LH$="Y" GOSUB 4810
4800 GOTO 4980
4810 FOR JX=1 TO JL
4820 IF I<=37 AND ID(JX)<37 THEN GOSUB 4870: GOTO 4850
4830 IF I>37 AND I<= 73 AND ID(JX)>=37 AND ID(JX)<73 THEN GOSUB 4870: GOTO 4850
4840 IF I>73 AND ID(JX)>=73 THEN GOSUB 4870
4850 NEXT JX
4860 RETURN
4870 FOR JZ=1 TO 10
4880 IF I<=37 THEN IP=ID(JX)
4890 IF I>37 AND I<=73 THEN IP=ID(JX)-36
4900 IF I>73 THEN IP=ID(JX)-72
4910 IF IP/3=INT(IP/3) THEN PP=(IP/3)*64 ELSE PP=(INT(IP/3)+1)*64
4920 IF IP/3=INT(IP/3) THEN PP=PP+42 : GOTO 4940
4930 IF (IP+1)/3 = INT((IP+1)/3) THEN PP=PP+21
4940 PRINT @PP,CHR$(143);: FOR JY=1 TO 25 : NEXT JY : PRINT @PP,"H";
4950 NEXT JZ
4960 RETURN
4970 PRINT @896,"Do you want to see the residuals?";
4980 PRINT @896,"Do you want to see the residuals?";
4990 KP=960 : L$="y" : R$="n" : GOSUB 360
5000 IF ZQ="Y" OR ZQ="y" THEN 6010
5010 PRINT @896, BL$ : PRINT @896,"Do you want to see a plot of the residuals?";
5020 GOSUB 360
5030 IF ZQ="Y" OR ZQ="y" THEN 6510
5040 GOTO 6910
```

<u>**DISPLAY RESIDUALS**</u>

```
6000 REM *** D I S P L A Y   R E S I D U A L S ***
6010 CLS
6020 PRINT" Y Observed              Y Estimate            Residual"
6030 FOR I=1 TO IN
6040 PRINT USING"#######.###";RV(I)*SQR(Y2)+YM;:PRINT"        ";:PRINT USING"###########.###";RH
(I);:PRINT"        ";:PRINT USING"#########.###";E(I)
6050 IF I/12 <> INT(I/12) GOTO 6100
6060 IF I=IN GOTO 6100
6070 PRINT:PRINT"PRESS SPACE BAR TO DISPLAY REMAINDER OF RESIDUALS"
6080 IF INKEY$ <> " " GOTO 6080
6090 CLS:PRINT @0," Y Observed              Y Estimate            Residual"
6100 NEXT I
6110 PRINT @896,"Do you want to see a plot of the residuals?";
6120 KP=960 : L$="y" : R$="n" : GOSUB 360
6130 IF ZQ="Y" OR ZQ="y" THEN 6510
6140 GOTO 6910
```

<u>**DISPLAY PLOT OF RESIDUALS**</u>

```
6500 REM *** P L O T   O F   R E S I D U A L S ***
6510 CLS:PRINT @0,"THE  HORIZONTAL AXIS  REPRESENTS  Y ESTIMATE
6520 PRINT @192,"THE  VERTICAL AXIS  REPRESENTS  RESIDUAL OR ERROR OF ESTIMATE
6530 YS=1E+25 : YB=1E-25 : RL=YS : RB=YB
6540 FOR I=1 TO IN
6550 IF RH(I)<YS THEN YS=RH(I)
6560 IF RH(I)>YB THEN YB=RH(I)
6570 IF E(I)<RL THEN RL=E(I)
6580 IF E(I)>RB THEN RB=E(I)
6590 NEXT I
6600 PRINT @64,"   RANGE OF Y ESTIMATE =";YS;"TO";YB
6610 PRINT @256,"   RANGE OF RESIDUALS =";RL;"TO";RB
6620 PRINT @768,"*** AFTER VIEWING THE PLOT, PRESS SPACE BAR TO CONTINUE"
```

```
6630 HU=118/(YB-YS) : VU=48/(RB-RL)
6640 IF YS>=0 THEN VA=10 ELSE VA=10+ABS(YS*HU)
6650 IF RL>=0 THEN HA=47 ELSE HA=47-ABS(RL*VU)
6660 PRINT @896,"PRESS SPACE BAR TO DISPLAY PLOT"
6670 IF INKEY$ <> " " GOTO 6670
6680 CLS
6690 FOR I=0 TO 47
6700 SET(VA,I)
6710 NEXT I
6720 FOR I=10 TO 127
6730 SET (I,HA)
6740 NEXT I
6750 K1=INT(VA/2)-5 : K2=960+INT(VA/2)-5 : K3=INT(HA/3+1.5) : IF HA/3 - INT(HA/3)>=2/3 THEN K3=K3-1
6760 IF K3>15 THEN K3=15
6770 PRINT @K1,"";: IF ABS(INT(RB+.5))>=1 THEN PRINT STR$(INT(RB+.5)); ELSE PRINT USING"#.###";RB;
6780 PRINT @K2,"";: IF ABS(INT(RL+.5))>=1 THEN PRINT STR$(INT(RL+.5)); ELSE PRINT USING"#.###";RL;
6790 IF VA<=14 AND YS<0 THEN K4=K3*64 ELSE K4=K3*64+6
6800 PRINT @K4,"";:IF ABS(INT(YS+.5))>=1 THEN PRINT STR$(INT(YS+.5)); ELSE PRINT USING"#.####";YS;
6810 IF ABS(INT(YB+.5))>=1 THEN PRINT @K3*64+63-LEN(STR$(INT(YB+.5))),STR$(INT(YB+.5)); ELSE PRINT @1018,"";:PRINT USING"#.###";YB;
6820 FOR I=1 TO IN
6830 O = 10+(RH(I)-YS)*HU : D = 47 - ( (E(I)-RL)*VU )
6840 IF O>127 THEN O=127
6850 IF D<0 THEN D=0
6860 IF O=INT(VA) THEN O=O+1
6870 IF D=INT(HA) THEN RESET(O-1,D): IF O<>127 THEN RESET(O+1,D)
6880 SET (O,D)
6890 NEXT I
6900 IF INKEY$ <> " " GOTO 6900
6910 PRINT @832, BL$ : PRINT BL$ :PRINT BL$;
6920 PRINT @832,"(1) RETURN TO ANOVA TABLE";
6930 IF CV>1 THEN PRINT @896,"(2) RERUN REGRESSION WITH A DIFFERENT SUBSET OF CARRIERS";
6940 IF CV>1 THEN PRINT @960,"(3)"; ELSE PRINT @960,"(2)";
6950 PRINT @964,"RETURN TO PROGRAM MENU";
6960 KP=768 : IF CV>1 KS=3 ELSE KS=2
6970 GOSUB 300
6980 IF IO=1 THEN 4010ELSE IF IO=2 AND CV=1 THEN 160ELSE IF IO=2 THEN 1500ELSE 160
```

PROGRAM SELECTION MENU

```
10000 REM *** P R O G R A M   S E L E C T I O N   M E N U ***
10010 CLS
10020 PRINT TAB(20)"PROGRAM SELECTION MENU"
10030 PRINT @128,"#1 Data I/O, Transformatons, Prob. & Inverse Prob. Distributions"
10040 PRINT"#2 Sample Stat's, 7-Num-Sum, Plots":PRINT
10050 PRINT"#3 Parametric Tests of Hypotheses and Confidence Intervals":PRINT
10060 PRINT"#4 Robust Estimates and Confidence Intervals":PRINT
10070 PRINT"#5 Multiple Linear Regression":PRINT
10080 PRINT"#6 Analysis of Variance"
10090 KP=896 : KS=6 : GOSUB 300
10100 IF IO<1 OR IO>6 PRINT @896,CHR$(250);:PRINT @896,"*** NOT A VALID OPTION ***";:FOR I=1 TO 1000:
NEXT : PRINT @896,CHR$(250);:GOTO10090
10110 IF IO=5 GOTO 30
10120 IF IO=6 THEN ZT="3" ELSE ZT="1"
10130 CLS
10140 IF IO<>6 THEN PRINT"From program #5 access to programs #1 thru #4 is through the   Control Module in
program #1." : PRINT
10150 PRINT"INSERT TAPE #";ZT;" INTO ESF-80 AND PRESS SPACE BAR"
10160 IF INKEY$<>" " GOTO 10160
10170 IF IO=6 THEN IO=2 ELSE IO=1
10180 IF IO=1 THEN PRINT : PRINT "Program #1 is now being loaded..."
10190 @LOAD IO
```

END OF PROGRAM #5

PROGRAM #6

MENU

```
5 CLEAR
10 CLEAR 200
20 CLS
30 DEFINT I-K : DEFSTR L
35 LB$=CHR$(123) : MB$=CHR$(176) : RB$=CHR$(125) : BL$=CHR$(252)
40 DIM PS(100), C(10,20)
50 PRINT TAB(20);"Analysis of Variance"
60 PRINT TAB(26);"(ANOVA)"
70 PRINT:PRINT"Menu:"
80 PRINT"   (1) One-Way Classification
90 PRINT"       (equal or unequal no. of observations for each factor)"
100 PRINT"  (2) Two-Way Classification
110 PRINT"      (with or without replication)
120 PRINT"  (3) Three-Way Classification
130 PRINT"  (4) Return to Program Selection Menu"
150 LW=CHR$(255): LP="######.###" : LD="##"
160 KP=896 : KS=4 : GOSUB 1000 : WS=IO : XZ=0
170 CLS : IF WS=4 THEN 20000
180 PRINT "Will Data be Entered via  ";LB$;"K";RB$;"eyboard  or  ";LB$;"T";RB$;"ape ?"; : KP=128 : L$="k" : R$
="t" : GOSUB 1200
190 IF Q$="K" OR Q$="k" THEN II=1 ELSE II=2 : GOSUB 6000
200 CLS
210 ON WS GOSUB 5000,10000,15000,20000
```

PRINT ANOVA TABLE

```
215 CLS
220 IF WS=1 LH="(One-Way Classification)":GOTO 260
230 IF WS=2 LH="(Two-Way Classification)":GOTO 260
240 IF WS=3 LH="(Three-Way Classification)"
260 PRINT TAB(15);"ANOVA ";LH
270 PRINT" SOURCE    D.F.   SUM OF SQUARES   MEAN SQUARE      F
280 PRINT STRING$(62,"-") : ON WS GOSUB 300,400,800
285 CLS : PRINT"Do you want to save the data to tape?"; : KP=37 : L$="y" : R$="n" : GOSUB 1200 : LQ=Q$
290 IF LQ<>"y" AND LQ<>"Y" AND LQ<>"n" AND LQ<>"N" THEN GOTO 285
295 IF LQ="y" OR LQ="Y" GOSUB 7000
297 CLS:GOTO 50
300 PRINT LC;TAB(13);CDF;TAB(19);:PRINT USING LP;TRTS;:PRINT TAB(36);:PRINT USING LP;MC;:PRINT TAB(5
0);:PRINT USING LP;F
310 PRINT
320 PRINT"Residual";TAB(13);EDF;TAB(19);:PRINT USING"######.###";ESS;:PRINT TAB(36);:PRINT USING"##
####.###";ME
330 PRINT
340 PRINT STRING$(62,"-")
350 PRINT"Total";TAB(13);TDF;TAB(19);:PRINT USING"######.###";TTS
360 PRINT @960,"Press SPACE BAR to continue";
370 IF INKEY$ <> " " GOTO 370
380 CLS : RETURN
400 IF IE>1 PRINT "Interaction";TAB(13);:PRINT USING LD;ADF;:PRINT TAB(19);:PRINT USING LP;TISS;:PRINT
 TAB(36);:PRINT USING LP;MA;:PRINT TAB(50);:PRINT USING LP;FA:PRINT
410 IF IE=1 PRINT @192,""; ELSE PRINT @320,"";
```

```
420 PRINT LR;TAB(13);:PRINT USING LD;RDF;:PRINT TAB(19);:PRINT USING LP;TRSS;:PRINT TAB(36);:PRINT U
SING LP;MR;:PRINT TAB(50);:PRINT USING LP;FR:PRINT
430 IF IE=1 PRINT @320,""; ELSE PRINT @448,"";
440 PRINT LC;TAB(13);:PRINT USING LD;CDF;:PRINT TAB(19);:PRINT USING LP;TCSS;:PRINT TAB(36);:PRINT U
SING LP;MC;:PRINT TAB(50);:PRINT USING LP;FC:PRINT
450 IF IE=1 PRINT @448,""; ELSE PRINT @576,"";
460 PRINT "Residual";TAB(13);:PRINT USING LD;EDF;:PRINT TAB(19);:PRINT USING LP;ESS;:PRINT TAB(36);:PR
INT USING LP;ME
470 PRINT STRING$(62,"-")
480 IF XI=1 GOTO 670
490 PRINT"Total";TAB(13);:PRINT USING LD;TDF;:PRINT TAB(19);:PRINT USING LP;TTS
500 IF IE=1 GOTO 670
510 FOR I=1 TO 25 : PRINT @253,CHR$(139) : FOR J=1 TO 25 : NEXT J : PRINT @253," " : FOR J=1 TO 25 : NEXT
J : NEXT I
520 PRINT @832,"REMINDER: a large F-statistic  => rejection"
530 FOR I=1 TO 1500 : NEXT I
540 PRINT @832, LW : PRINT LW
550 PRINT @832,"Do you accept or reject the hypothesis that there is no inter-";PRINT"action between the row a
nd column factors?";:KP=936 : L$="a" : R$="r" : GOSUB 1200 : LQ=Q$
570 PRINT @832,LW : PRINT LW
580 IF LQ="a" OR LQ="A" GOTO 620
590 PRINT @768,"Since you reject the hypothesis of no interaction the linear";PRINT"model is violated and the re
mainder of the ANOVA is invalid":PRINT"***      Press space bar to continue";
600 IF INKEY$ <> " " GOTO 600
610 CLS:RETURN
620 PRINT @832, LW : PRINT LW
630 PRINT @192, LW : PRINT @192,"Hypothesis that there is no interaction is accepted"
640 PRINT @832,"Do you want to pool interaction & residual?"; L$="y" : R$="n" : KP=873 : GOSUB 1200 : LQ=Q$
660 IF LQ = "y" OR LQ ="Y" GOTO 700
670 PRINT @768, LW : PRINT LW : PRINT LW : PRINT @768,LR;" & ";LC;" hypotheses can be assessed using" : PR
INT"the F-ratios shown above."; PRINT"        ( Press space bar to continue)";
680 IF INKEY$ <> " " GOTO 680
690 CLS : RETURN
700 ESS=TISS + ESS : EDF=ADF+EDF : ME=ESS/EDF : FC=MC/ME : FR=MR/ME : XI=1 : GOTO 410
800 PRINT LR;TAB(13);:PRINT USING LD;RDF;:PRINT TAB(19);:PRINT USING LP;TRSS;:PRINT TAB(36);:PRINT U
SING LP;MR;:PRINT TAB(50);:PRINT USING LP; F1
810 PRINT LC;TAB(13);:PRINT USING LD;CDF;:PRINT TAB(19);:PRINT USING LP;TCSS;:PRINT TAB(36);:PRINT U
SING LP;MC;:PRINT TAB(50);:PRINT USING LP; F2
820 PRINT LG;TAB(13);:PRINT USING LD;GDF;:PRINT TAB(19);:PRINT USING LP;TGSS;:PRINT TAB(36);:PRINT U
SING LP;MG;:PRINT TAB(50);:PRINT USING LP; F3
830 PRINT : PRINT"Interactions:"
840 PRINT LEFT$(LR,5);"-";LEFT$(LC,5);TAB(13);:PRINT USING LD;DDF;:PRINT TAB(19);:PRINT USING LP;DOSS
;:PRINT TAB(36);:PRINT USING LP;MD;:PRINT TAB(50);:PRINT USING LP; F4
850 PRINT LEFT$(LR,5);"-";LEFT$(LG,5);TAB(13);:PRINT USING LD;BDF;:PRINT TAB(19);:PRINT USING LP;DBSS
;:PRINT TAB(36);:PRINT USING LP;MB;:PRINT TAB(50);:PRINT USING LP;F5
860 PRINT LEFT$(LC,5);"-";LEFT$(LG,5);TAB(13);:PRINT USING LD;ADF;:PRINT TAB(19);:PRINT USING LP;AISS
;:PRINT TAB(36);:PRINT USING LP;MA;:PRINT TAB(50);:PRINT USING LP; F6
870 PRINT : PRINT"Residual";TAB(13);:PRINT USING LD;EDF;:PRINT TAB(19);:PRINT USING LP;ESS;:PRINT TAB
(36);:PRINT USING LP; ME
880 PRINT STRING$(63,"-")
890 PRINT "Total";TAB(13);:PRINT USING LD;TDF;:PRINT TAB(19);:PRINT USING LP;TSS
900 PRINT @960,"*** Press SPACE BAR to return to continue";
910 IF INKEY$<>" " GOTO 910
920 CLS
930 RETURN
```

<u>OPTION SELECTION</u>

```
1000 PRINT @KP+5,"OPTION DESIRED ";LB$;MB$;RB$;
1010 I$=INKEY$ : IF I$="" THEN 1010 ELSE IO=VAL(I$)
1020 PRINT @KP+21, I$; : FOR I=1 TO 100 : NEXT
1030 IF IO<1 OR IO>KS THEN PRINT @KP, BL$;: PRINT @KP,"***  NOT A VALID OPTION  ***";: FOR I=1 TO 100
0 : NEXT : PRINT @KP, BL$;: GOTO 1000
```

```
1040 RETURN
1200 PRINT @KP+5,LB$;" ";L$;" or ";R$;" ";RB$;" ";LB$;MB$;RB$;
1210 Q$=INKEY$ : IF Q$="" THEN 1210 ELSE PRINT @KP+20,Q$;: FOR I=1 TO 100 : NEXT
1220 IF Q$<>L$ AND Q$<>CHR$(ASC(L$)-32) AND Q$<>R$ AND Q$<>CHR$(ASC(R$)-32) THEN 1200
1230 RETURN
```

ONE-WAY CLASSIFICATION

```
5000 REM *** ONE-WAY CLASSIFICATION
5005 IF II=2 THEN ON QL GOTO 5110, 5390
5010 PRINT TAB(27);"ANOVA" : PRINT TAB(17);"(One-Way Classification)"
5020 PRINT @128,"Enter a reference name for the factor (i.e. Operator":INPUT"(enter up to 12 characters)";LC
5030 PRINT @128,LW : PRINT LW
5040 PRINT @128,"Enter the number of ";LC;" levels (up to 10)";:INPUT IC
5050 PRINT @129,LW : PRINT @128,"Are there an equal number of experiments / observations" : PRINT"for each level?";
5052 KP=205 : L$="y" : R$="n" : GOSUB 1200 : LQ=Q$
5055 IF LQ="y" OR LQ="Y" THEN QL=1 ELSE QL=2
5060 PRINT @128,LW : PRINT LW
5080 IF LQ="n" OR LQ="N" GOTO 5390
5090 PRINT @128,LW : PRINT LW
5100 PRINT @128,"Enter number of observations for each ";LC;" (up to 20)":INPUT IE
5110 S=0 : OS=0
5115 CLS : IF II=2 THEN PRINT @0,"Calculating"
5120 FOR I=1 TO IC
5130    IF II=1 THEN PRINT LC;I
5140    CS(I)=0
5150    FOR J=1 TO IE
5160       IF II=1 THEN PRINT"  Enter value of observation";J;:INPUT C(I,J)
5170       CS(I)=CS(I)+C(I,J)
5180       S=S+C(I,J)[2
5190       OS=OS+C(I,J)
5200    NEXT J
5210    IF II=1 THEN PRINT
5220 NEXT I
5230 CLS : PRINT @0,"Calculating"
5240 OS=OS[2/(IC*IE)
5250 TRTS=0
5260 FOR I=1 TO IC
5270    TRTS=TRTS+CS(I)[2
5280 NEXT I
5290 TRTS=(TRTS/IE)-OS
5300 TTS=S-OS
5310 ESS=TTS-TRTS
5320 CDF=IC-1
5330 EDF=IC*(IE-1)
5340 TDF=IC*IE-1
5350 MC=TRTS/CDF
5360 ME=ESS/EDF
5370 F=MC/ME
5380 CLS : RETURN
5390 CLS : IF II=2 THEN PRINT @0,"Calculating"
5400 OS=0 : S=0 : COB=0 : JN=0
5410 FOR I=1 TO IC
5420 IF II=1 THEN PRINT"Enter the number of observations for ";LC;I;"(up to 20)":INPUT IX(I)
5430 CS(I)=0 : CB(I)=0 : JN=JN+IX(I)
5440    FOR J=1 TO IX(I)
5450       IF II=1 THEN PRINT"For ";LC;I;" enter observation";J;:INPUT C(I,J)
5460       CS(I)=CS(I)+C(I,J)
5470       OS=OS+C(I,J)
5480       S=S+C(I,J)[2
5490    NEXT J
```

```
5500    PRINT
5510 NEXT I
5520 CLS : PRINT @0,"Calculating"
5525 TRTS=0
5530 FOR I=1 TO IC
5540    TRTS=TRTS+CS(I)(2/IX(I)
5550 NEXT I
5560 TRTS=TRTS-OS(2/JN
5570 TTS=S-OS(2/JN
5580 ESS=TTS-TRTS
5590 CDF=IC-1
5600 EDF=JN-IC
5610 TDF=CDF+EDF
5620 MC=TRTS/CDF
5630 ME=ESS/EDF
5640 F=MC/ME
5650 CLS : RETURN
```

LOAD DATA FROM TAPE

```
6000 REM *** L O A D  D A T A  F R O M  T A P E ***
6010 CLS
6020 PRINT"NOTE: You must specify a file number containing data which was"
6030 PRINT"recorded in the format for ";
6040 IF WS=1 PRINT"one-way classification." : GOTO 6100
6050 IF WS=2 THEN PRINT"two-way classification." ELSE PRINT "three-way classification"
6100 PRINT : PRINT"Insert Data Tape into ESF-80 and press SPACE BAR"
6110 IF INKEY$<>" " GOTO 6110
6120 PRINT:INPUT "Enter File # of Data to be Loaded";JF
6130 @CLOSE
6140 @OPEN JF
6150 SS=WS
6160 @INPUT WS
6170 IF WS=SS GOTO 6300
6180 CLS : PRINT"ERROR... the data file you are attempting to load is for "
6190 LL(1)="one" : LL(2)="two" : LL(3)="three"
6200 PRINT LL(WS);"-way classification instead of ";LL(SS);"-way classification!"
6210 PRINT : PRINT"Do you want to continue to load data and run a ";LL(WS);"-way"
6220 PRINT"classification ANOVA or do you want to start over?"
6230 PRINT @896,"";:INPUT"  (1) Continue    (2) Start Over    ( 1 or 2 )";IO
6240 IF IO<>1 AND IO<>2 PRINT @896, LN : PRINT @896,"*** NOT A VALID OPTION ***": FOR I=1 TO 1000: NE
XT : GOTO 6230
6250 IF IO=2 RUN
6300 ON WS GOTO 6310, 6500, 6700
6310 @INPUT LC
6320 @INPUT QL
6330 @INPUT IC
6340 IF QL=2 GOTO 6400
6350 @INPUT IE
6360 FOR I=1 TO IC
6370 FOR J=1 TO IE
6380 @INPUT C(I,J)
6385 NEXT J
6390 NEXT I
6395 @CLOSE
6397 RETURN
6400 FOR I=1 TO IC
6410 @INPUT IX(I)
6420 FOR J=1 TO IX(I)
6430 @INPUT C(I,J)
6440 NEXT J
6450 NEXT I
```

```
6460 @CLOSE
6470 RETURN
6500 @INPUT LR,LC
6510 @INPUT IR,IC,IE
6520 FOR I=1 TO IR
6530 FOR J=1 TO IC
6540 FOR K=1 TO IE
6550 @INPUT X(I,J,K)
6560 NEXT K
6570 NEXT J
6580 NEXT I
6590 @CLOSE
6600 RETURN
6700 @INPUT LR,LC,LG
6710 @INPUT IX,IR,IC,IG
6720 FOR I=1 TO IR
6730 FOR J=1 TO IC
6740 FOR K=1 TO IG
6750 @INPUT X(I,J,K)
6760 NEXT K
6770 NEXT J
6780 NEXT I
6790 @CLOSE
6800 RETURN
```

SAVE DATA TO TAPE

```
7000 REM *** S A V E  D A T A  T O  T A P E ***
7005 CLS
7010 PRINT"Data will be recorded in accordance with the format of"
7020 PRINT"ANOVA ";LH
7030 PRINT : PRINT"Insert Data Tape into ESF-80 and press SPACE BAR"
7040 IF INKEY$ <> " " GOTO 7040
7050 PRINT : INPUT"Enter File # under which you want data recorded";JF
7055 @CLOSE
7060 @OPEN JF
7070 @PRINT WS
7080 ON WS GOTO 7100, 7400, 7700
7100 @PRINT LC
7110 @PRINT QL
7120 @PRINT IC
7130 IF QL=2 GOTO 7250
7140 @PRINT IE
7150 FOR I=1 TO IC
7160 FOR J=1 TO IE
7170 @PRINT C(I,J)
7180 NEXT J
7190 NEXT I
7200 @CLOSE
7210 RETURN
7250 FOR I=1 TO IC
7255 @PRINT IX(I)
7260 FOR J=1 TO IX(I)
7270 @PRINT C(I,J)
7280 NEXT J
7290 NEXT I
7300 @CLOSE
7310 RETURN
7400 @PRINT LR,LC
7410 @PRINT IR,IC,IE
7420 FOR I=1 TO IR
7430 FOR J=1 TO IC
```

```
7440 FOR K=1 TO IE
7450 @PRINT X(I,J,K)
7460 NEXT K
7470 NEXT J
7480 NEXT I
7490 @CLOSE
7500 RETURN
7700 @PRINT LR,LC,LG
7710 @PRINT IX,IR,IC,IG
7720 FOR I=1 TO IR
7730 FOR J=1 TO IC
7740 FOR K=1 TO IG
7750 @PRINT X(I,J,K)
7760 NEXT K
7770 NEXT J
7780 NEXT I
7790 @CLOSE
7800 RETURN
```

<u>TWO - WAY CLASSIFICATION</u>

```
10000 REM  *** TWO-WAY CLASSIFICATION
10005 IF II=2 THEN GOTO 10070
10010 PRINT TAB(27);"ANOVA":PRINT TAB(17);"(Two-Way Classification)"
10020 PRINT @192,"Enter a reference name for the row factor (up to 12 characters)":INPUT LR
10030 PRINT @192, LW : PRINT,LW : PRINT @192,"Enter the number of ";LR;" levels";:INPUT IR
10040 PRINT @192, LW : PRINT @192,"Enter a reference name for the column factor (up to 12 char.'s)":INPUT LC
10050 PRINT @192, LW : PRINT LW : PRINT @192,"Enter the number of ";LC;" levels";:INPUT IC
10055 PRINT @192, LW
10060 PRINT @192,"Enter the number of replications in each cell (up to 10)";:INPUT IE
10070 CLS : IF IE <> 1 PRINT "The model is a :" ELSE GOTO 10130
10080 PRINT"  (1) Fixed Effects Model
10090 PRINT"  (2) Random Effects Model
10100 PRINT"  (3) Rows Fixed, Columns Random
10110 PRINT"  (4) Columns Fixed, Rows Random
10120 KP=@96 : KS=4 : GOSUB 1000 : JM=IO
10130 CLS : IF IE=1 THEN JM=1
10135 IF II=2 PRINT @0,"Calculating"
10140 S=0 : OS=0
10150 FOR I=1 TO IR : RS(I)=0
10160 IF II=1 THEN PRINT"For ";LR;I
10170 FOR J=1 TO IC
10180 IF II=1 THEN PRINT"   For ";LC;J
10190 IJ=(I-1)*IC + J : PS(IJ)=0
10200 FOR K=1 TO IE
10210 IF II=1 THEN PRINT"       enter replication #";K;:INPUT X(I,J,K)
10220 Y=X(I,J,K)
10230 S=S+Y[2 : OS=OS+Y : RS(I)=RS(I)+Y : PS(IJ)=PS(IJ)+Y
10240 NEXT K
10250 NEXT J
10255 IF II=1 THEN CLS
10260 NEXT I
10270 CLS : PRINT"Calculating"
10280 OS=OS[2/(IR*IC*IE)
10290 FOR J=1 TO IC
10300 CS(J)=0
10310 FOR I=1 TO IR
10320 FOR K=1 TO IE
10330 CS(J)=CS(J)+X(I,J,K)
10340 NEXT K
10350 NEXT I
10360 NEXT J
```

```
10370 TRSS=0 : TCSS=0 : TISS=0
10380 FOR I=1 TO IR : TRSS=TRSS+RS(I)[2 : NEXT I : TRSS=TRSS/(IC*IE)-OS : RDF=IR-1 : MR=TRSS/RDF
10390 FOR J=1 TO IC : TCSS=TCSS+CS(J)[2 : NEXT J : TCSS=TCSS/(IR*IE)-OS : CDF=IC-1 : MC=TCSS/CDF
10400 IF IE=1 ADF=0 : TISS=0 : GOTO 10420
10410 FOR K=1 TO IC*IR : TISS=TISS+PS(K)[2 : NEXT K : TISS=TISS/IE-OS-TRSS-TCSS : ADF=RDF*CDF : MA=TI
SS/ADF
10420 TTS=S-OS : TDF=IR*IC*IE-1
10430 ESS=TTS-TRSS-TCSS-TISS : EDF=TDF-ADF-CDF-RDF : ME=ESS/EDF
10440 IF IE=1 MA=ME
10450 ON JM GOTO 10460, 10480, 10500, 10520
10460 FR=MR/ME : FC=MC/ME : FA=MA/ME
10470 GOTO 10530
10480 FR=MR/MA : FC=MC/MA : FA=MA/ME
10490 GOTO 10530
10500 FR=MR/MA : FC=MC/ME : FA=MA/ME
10510 GOTO 10530
10520 FR=MR/ME : FC=MC/MA : FA=MA/ME
10530 CLS : RETURN
```

__THREE - WAY CLASSIFICATION__

```
15000 REM *** T H R E E  W A Y  C L A S S I F I C A T I O N ***
15002 IF II=2 THEN GOTO 15085
15005 IX=0
15010 PRINT TAB(27)"ANOVA" : PRINT TAB(16)"(Three-Way Classification)"
15020 PRINT @192,"Enter a refrence name for the row factor (up to 12 characters)";INPUT LR
15030 PRINT @192, LW :PRINT LW
15040 PRINT @192,"Enter the number of ";LR;" levels (up to 10)";:INPUT IR
15045 IF IR>IX THEN IX=IR
15050 PRINT @192,LW : PRINT @192,"Enter a reference name for the column factor (up to 12 char.'s)";INPUT LC
15060 PRINT @192, LW : PRINT LW : PRINT @192,"Enter the number of ";LC;" levels (up to 10)";:INPUT IC
15065 IF IC>IX THEN IX=IC
15070 PRINT @192, LW : PRINT LW : PRINT @192,"Enter a reference name for the group factor (up to 12 char.'s) "
:INPUT LG
15080 PRINT @192, LW : PRINT LW : PRINT @192,"Enter the number of ";LG;" levels (up to 10)";:INPUT IG
15082 IF IG>IX THEN IX=IG
15085 CLS : S=0 : OS=0 : IF II=2 THEN PRINT"Calculating"
15090 FOR I=1 TO IX
15095 RS(I)=0 : CS(I)=0 : GS(I)=0
15100 FOR J=1 TO IX
15105 RC(I,J)=0 : RG(I,J)=0 : CG(I,J)=0
15110 NEXT J
15115 NEXT I
15130 FOR I=1 TO IG
15140 IF II=1 THEN PRINT"For ";LG;I
15150 FOR J=1 TO IR
15160 IF II=1 THEN PRINT"  For ";LR;J
15180 FOR K=1 TO IC
15190 IF II=1 THEN PRINT"     Enter observation for ";LC;K;:INPUT X(J,K,I)
15200 Y=X(J,K,I)
15220 S=S+Y[2 : OS=OS+Y
15230 RS(J)=RS(J)+Y : CS(K)=CS(K)+Y : GS(I)=GS(I)+Y
15240 RC(J,K)=RC(J,K)+Y : RG(J,I)=RG(J,I)+Y : CG(K,I)=CG(K,I)+Y
15250 NEXT K
15260 NEXT J
15265 IF II=1 THEN CLS
15270 NEXT I
15280 CLS : PRINT"Calculating"
15290 OS=OS[2/(IR*IC*IG)
15300 TRSS=0 : TCSS=0 : TGSS=0
15310 FOR I=1 TO IR : TRSS=TRSS+RS(I)[2 : NEXT I : TRSS=TRSS/(IC*IG) - OS : RDF=IR-1 : MR=TRSS/RDF
15320 FOR J=1 TO IC : TCSS=TCSS+CS(J)[2 : NEXT J : TCSS=TCSS/(IR*IG) - OS : CDF=IC-1 : MC=TCSS/CDF
```

```
15330 FOR K=1 TO IG : TGSS=TGSS+GS(K)[2 : NEXT K : TGSS=TGSS/(IR*IC) - OS : GDF=IG-1 : MG=TGSS/GDF
15340 DOSS=0 : DBSS=0 : AISS=0
15380 FOR I=1 TO IX
15390 FOR J=1 TO IX
15400 DOSS=DOSS+RC(I,J)[2 : DBSS=DBSS+RG(I,J)[2 : AISS=AISS+CG(I,J)[2
15410 NEXT J
15420 NEXT I
15430 DOSS=DOSS/IG - OS - TRSS - TCSS : DDF=(IR-1)*(IC-1) : MD=DOSS/DDF
15440 DBSS=DBSS/IC - OS - TRSS - TGSS : BDF=(IR-1)*(IG-1) : MB=DBSS/BDF
15450 AISS=AISS/IR - OS - TCSS - TGSS : ADF=(IC-1)*(IG-1) : MA=AISS/ADF
15455 TSS=S-OS : TDF=IR*IC*IG-1
15460 ESS=TSS-TRSS-TCSS-TGSS-DOSS-DBSS-AISS : EDF=TDF-RDF-CDF-GDF-DDF-BDF-ADF : ME=ESS/EDF
15470 F1=MR/ME
15480 F2=MC/ME
15490 F3=MG/ME
15500 F4=MD/ME
15510 F5=MB/ME
15520 F6=MA/ME
15530 RETURN
```

PROGRAM SELECTION MENU

```
20000 REM *** P R O G R A M  S E L E C T I O N  M E N U ***
20010 CLS
20020 PRINT TAB(20)"PROGRAM SELECTION MENU"
20030 PRINT @128,"#1 Data I/O, Transformations, Prob. & Inv. Prog. Distributions":PRINT
20040 PRINT"#2 Sample Stat's, 7-Num Sum, Plots":PRINT
20050 PRINT"#3 Parametric Tests of Hypotheses and Confidence Intervals":PRINT
20060 PRINT"#4 Robust Estimates and Confidenct Intervals":PRINT
20070 PRINT"#5 Multiple Linear Regression":PRINT
20080 PRINT"#6 Analysis of Variance"
20090 KP=896 : KS=6 : GOSUB 1000
20110 IF IO=6 GOTO 5
20120 IF IO=5 THEN LT="3" ELSE LT="1"
20125 CLS
20130 IF IO<>5 THEN PRINT"From program #6 access to programs #1 thru #4 is thorugh the   Control Module in program #1." : PRINT
20150 IO=1
20170 PRINT"INSERT TAPE# ";LT;" INTO ESF-80 AND PRESS SPACE BAR"
20180 IF INKEY$ <> " " GOTO 20180
20185 IF LT<>"3" THEN PRINT"Program #1 now being loaded..."
20190 @LOAD IO
```

END OF PROGRAM #6

LIST OF REFERENCES

1. McNeil, D. R., Interactive Data Analysis, p. vi, Wiley, 1977

2. DeGrott, M. H., Probability and Statistics, p. 437 Addison-Wesley, 1975

3. McNeil, p. 12

4. Kennedy, W. J. and Gentle, J. E., Statistical Computing, p. 65, Dekker, 1980

5. Handbook of Tables for Probability and Statistics, 2nd ed., CRC Press, 1968

6. Abramowitz M. and Stegun, I. A., Handbook of Mathematical Functions, p. 932, National Bureau of Standards, 1964

7. Ibid., p. 941

8. Poole, L. and Brochers, M., Some Common BASIC Programs, p. 130, Osborne, 1977

9. Abramowitz and Stegun, p. 948-949

10. Kennedy and Gentle, p. 112

11. Abramowitz and Stegun, p. 932-949

12. Association for Computing Machinery,"Collected Algorithms from ACM", 1980

13. Abramowitz and Stegun, p. 947

14. McNeil, p. 5

15. Dixon, W. J. and Massey, F. J., Introduction to Statistical Analysis, 3rd ed., p. 330, McGraw-Hill, 1969

16. Ibid, p. 330-332

17. Mosteller, F. and Tukey, J. W., Data Analysis and Regression,

18. Dixon and Massey, p. 562

19. Draper, N. R. and Smith, H., Applied Regression Analysis, p. 144-150

20. Ibid., p. 1-35

21. Intriligator, M. D., Econometric Models, Techniques, & Applications, p. 124-127, Prentice-Hall, 1978

22. Hoaglin, D. C. and Welsch, R. E., "The Hat Matrix in Regression and ANOVA", The American Statistician, v.32, p. 17-22, Febreary 1978

23. McNeil, p. 51

24. Draper and Smith, p. 365-402

25. Duncan, A. J., Quality Control and Industrial Statistics, 4th ed., p. 609-684, Irwin, 1974

26. Morgeson, J. D., Statistical Programs for the Apple II Plus Microcomputer, Masters Thesis, Naval Postgraduate School, 1981

27. Duff, R. H., A Microcomputer-Based Network Optimization Package, Masters Thesis, Naval Postgraduate School, 1981

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
4.	LCDR C. F. Taylor, Code 55Ta Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
5.	Professor D. C. Barr, Code 55Bn Department of Operations Research Naval Postgraduste School Monterey, California 93940	1
6.	Professor D. C. Boger, Code 54Bk Department of Administrative Science Naval Postgraduate School Monterey, California 93940	1
7.	Maj R. P. Isbell, USMC Marine Corps Tactical Systems Support Activity (MCTSSA) Marine Corps Base, Camp Pendleton, California 92055	1
8.	Maj R. H. Duff, USMC First Marine Aircraft Wing FPO San Francisco, California 96602	1
9.	LCDR D. R. Dean, USN PRECOMM Unit USS Ticonderoga (CG-47) FPO New York, New York 09501	1

10. Professor R. H. Shudde, Code 55Su — 1
Department of Operations Research
Naval Postgraduate School
Monterey, California 93940

11. Capt. J. N. Lott, Code 30 — 1
Department of Operations Research
Naval Postgraduate School
Monterey, California 93940

12 LCDR J. R. Sartoris, Code 30 — 1
Department of Operations Research
Naval Postgraduate School
Monterey, California 93940