1971

AN EASY ACCESS INTERACTIVE STATISTICAL SYSTEM FOR USE AND TRAINING IN BIOMETRY

Amos Addison Slaymaker Jr.

AN EASY ACCESS INTERACTIVE STATISTICAL SYSTEM FOR USE AND TRAINING IN BIOMETRY

by

Amos Addison Slaymaker, Jr.

B.S., Lynchburg College, 1968

Thesis

submitted in partial fulfillment of the requirements for the

Degree of Master of Science in the Department of

Biometry at the Medical College of Virginia

Richmond, Virginia

June, 1971

This thesis by Amos Addison Slaymaker, Jr. is accepted in its present form as satisfying the thesis requirement for the degree of Master of Science

Date: Approved:

October 14, 1970 [redacted]

Advisor, Chairman of Graduate Committee

October 14, 1970 [redacted]

October 14, 1970 [redacted]

October 14, 1970 [redacted]

APPROVED: [redacted]

Dean of the School of Graduate Studies

CURRICULUM VITAE

[redacted]

ACKNOWLEDGEMENTS

The author wishes to express his sincere appreciation to Dr. Walter H. Carter, Jr. for suggesting this research and for his guidance during the course of this investigation. The author also extends gratitude to Dr. Raymond H. Myers for his assistance and encouragement. Gratitude is also extended to Dr. S. James Kilpatrick, Chairman of the Biometry Department, for his support and encouragement.

The author is indebted to his wife, Mrs. Carroll Slaymaker, for her excellent work in typing this thesis.

This investigation was supported in part by Public Health Service Research Grant RR 0016-08, from the National Institutes of Health.

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

One of the most important tools of the applied statistician is the digital computer. It is natural, therefore, for the instructor in applied statistics to want his students to become familiar with the use of computers. If his students are going to get actual experience in using a computer for statistical analysis, he often has only two alternatives. The students can be required to write their own statistical programs or they can use programs already available through a computer facility.

If the course is to be taught such that each student is responsible for his own programs, the instructor must either require that the students have previous programming experience or he must be prepared to spend a portion of his class time teaching a programming language. Neither of these seem to be satisfactory. First, to make knowledge of programming a prerequisite will often reduce the number of people interested in the course. Many students, who would otherwise enroll, might be completely unfamiliar with programming and have no real interest in becomming programmers. To spend a portion of the class time in teaching a programming language and associated programming techniques would often mean that the emphasis of the class could easily shift from the statistical methods to computer programming. This would result in a significant reduction in the amount of material the class could cover.

The alternative to having each student write his own programs is to use prepared programs available through a computer facility. In most instances, this would mean that each time a student wished to use the computer for a statistical analysis he would have to prepare the data for card input, send the cards to the computer facility, wait, and finally have his results returned. Again either the instructor would have to assign a particular program and would lead the class through the data preparation or he would expect each student to be responsible for reading the program documentation and preparing the data for himself. In many statistical analyses the investigator might wish to run several different programs. For each of these the student might have to review the relevant documentation, punch a new set of data cards and wait. Unfortunately, rather than repeat this procedure several times a student may become satisfied with running only the primary analysis without spending time, for instance, verifying the underlying assumptions.

An example of the type of situation which might indicate several computer runs would be data on which an Analysis of Variance is to be performed. Consider the problem of a student who has data from patients being treated with several different drugs. He wishes to test the null hypothesis of no significant differences between the treatment means. He might first wish to run a Bartlett's test for homogeneity of variances. If transforms are necessary on the data he will wish to try them. If he is satisfied that the variances are not significantly different, he will compute the Analysis of Variance possibly following that with Duncan's multiple range test. Since each method is probably done by a different program, the data might have to be completely punched three or four different times. Rather than doing all the extra work the student might simply run the Analysis of Variance and be satisfied with a less than a complete data analysis.

The problems introduced here give the necessary background for the discussion of the APL Statistical System which follows. This discussion is divided into three sections. The first section includes two chapters and discusses broadly the APL Statistical System characteristics which contribute to overcome some of the problems involved in utilizing a computer in statistical instruction. The second chapter describes two basic utilizations of the Statistical System.

The second section describes the computer hardware configuration on which the system is currently being implemented. It also describes some óf the important characteristics of the programming language used. A description of the actual statistical System with a list of the statistical methods which are available to the user is also included in the third chapter.

The third section is actually a user's manual giving the operating procedures for the system, an explanation of the keyboard, data entry, and a few of the basic APL operators. To make it an independent part of the thesis so that it may be used alone as a manual, a more complete description of how to use each of the statistical methods is given. For each method an example is shown which can be verified in most cases by the reference source listed in the example. A complete program listing of all the programs, or functions, used in this system can be found in the Appendix.

CHAPTER II

SYSTEM DESCRIPTION

Recall that in the introduction two alternative methods were discussed for introducing students to the use of computers in statistical analyses. Since major drawbacks existed in implementing either of these alternatives, it was felt that some type of compromise between the two might be found. If a realistic substitute were to be designed, two major conditions would have to be met. The first was that the system should allow the student immediate access to the computer. One way to satisfy this condition was to have a terminal connected directly to a computer.

The second condition is an extension of the first. Since having the student write his own programs is obviously unsatisfactory, it was felt that a library of methods would have to be available which would be easier to use and more meaningful than other prepared programs. One possibility was that the programs be so written as to require student interaction during the actual execution of the function. The user would be required to converse with the computer through a terminal until the program had the needed information to finish its calculations.

The APL Statistical System was primarily designed to satisfy these two conditions. In considering the language in which the system of programs would be written it was felt that APL (A Programming Language) was the logical choice. Many of the characteristics which the programs have are directly related to this programming language and are discussed in Chapters III & IV.

The APL Statistical System is a terminal oriented system of programs which allows the user to call on any (or all) of the system programs at any time. The user enters his data into the computer through the terminal or by cards and is able to load and execute any program in the system library. As the output from one function is studied the user can be operating on the same set of data with another of the library functions.

The actual data entry is accomplished by entering the data an element at a time, skipping a space between each data point. The format is always the same and each program is written such that the user is supplied with inquiries and a choice of possible responses as the program is executing. This means that there is a minimum of documentation which the user must read before actually performing the statistical analysis. He, of course, also has no need for learning any programming language. With the short introduction to the system presented in the User's Manual, a student or any user familiar with the statistical methods, should be able to execute any of the library functions.

Since this statistical system eliminates the traditional problems of data preparation, the student can concentrate more on the types of information which result from the various statistical analyses. Without having to become familiar with any programming language or with the involved documentation accompanying "canned" programs, the student is able to transform data, execute functions, and even use the computer as a sophisticated calculator. In other words, all the power of the computer is given to the student without any of the problems associated with its use. As a result the student is able to learn the extremely important role the computer can play in data analysis.

Another important consideration of this interactive system is that a particular program does not become a "black box", which the student must only assume is doing a particular method. At any time a listing of the whole function can be obtained from the computer. This display lists each of the statements in the function in the order of execution. As a result of the characteristics of the language, APL, the function statements have a very close resemblance to the algebraic statements in the student's source material. Even the summation of terms and the matrix and vector multiplications are written as one statement and are not lost within the program in complicated programming techniques. The student can, if he knows the basic symbols used in APL, see a one-to-one correspondence between what is happening within the displayed program and what are in his course materials. This additional knowledge of APL is not necessary for the normal use of the system.

The system of programs developed in this thesis has the following characteristics:

1. No knowledge of programming required
2. One simple data input format
3. Availability of all the programs
4. Allows for interactive mode in program execution

Therefore, the APL Statistical System appears to provide the needed compromise between student programming and the available "canned" programs.

CHAPTER III

SYSTEM UTILIZATION

The author visualizes two essentially different uses for the APL Statistical System: Academic and Consulting.

For the graduate student concentrating in statistics the system would be utilized with the course material. This would involve exercises and assignments from the instructor of the course. The system is designed only to do the heavy work for the student and not to give him all the final answers. For instance, the Analysis of Variance programs only give the Sum of Squares partitioned for the different effects and the degrees of freedom associated with each. The student is still responsible for forming the appropriate Mean Squares and F-ratios. If he is not familiar with the concepts for forming Expected Mean Squares the user will usually obtain an incorrect result. Thus, this system places emphasis on the method used by the student in arriving at his conclusions, not on having arrived at the correct answer.

For the graduate student not majoring in statistics it is often necessary to take a separate course in the introduction to statistics. A good example is the Medical student. With his primary interest in medicine he is often neither equipped nor interested in delving into the sophisticated techniques of statistical analysis. Because of this lack of background and interest he is seldom introduced to anything that would give him a feeling for the power of statistics. Far too often the impression is just the opposite, that statistics is an intuitively obvious discipline lacking in any imagination and sophistication. This is a result of the low level at which statistics is usually taught to such students.

A medical student is trained to diagnose a disease; similarly, he could be trained to "diagnose" a statistical analysis for a set of data. If the approach of the instructor was to give the student a check list of assumptions and conditions under which specific statistical methods could be used the student could, with the use of an interactive system, utilize many of the statistical techniques without ever having to become deeply involved with the calculations. The role of a graduate department is not to make theoretical statisticians of the students from outside the department. Its goal should be to introduce such a student to as many methods as possible, always insuring that the student is aware of the conditions which must be met before the analysis is run and the limits on the conclusions which can be reached afterwards. This approach would give the student an introduction to the scope of statistics and hopefully make him aware that there is still much he does not know. The student would realize that he should seek the advice of professional statisticians before, during, and after experiments in his own field. In that way statistics could be used more effectively in helping him arrive at his conclusions.

Frequently problems arise which require a statistical consultant. If the consultant, who is approached with the problem, is fortunate he will have a full computer facility available to him staffed by programmers and technicians waiting to help him use the computer for his statistical analysis. Unfortunately this is often not the case and the consultant is sometimes forced to spend as much time as the student coding data for input and decoding output for each program he wishes to use. Having the appropriate interactive system available to him means the consultant can do the analysis quickly and efficiently and feel confident that he has satisfied all assumptions and run all the tests appropriate for the data analysis. He is no longer dependent on the computer staff, but rather has a tool available which he can freely use without the need for introducing the consulting problem to a programmer.

It is often the case that a somewhat unusual problem will need data analysis. Very often this means that an existing program cannot be used because of some special condition which the program cannot handle. With other computer languages modifying a program could be a frustrating task because of the different programming techniques used by different programmers. Often the original author of the program is not available to help with the modification. This system of programs, however, is designed such that a modification by someone familiar with the programming language would be easy and depending on the amount of modification involved would mean the loss of very little time. The point is that modifications can even be made by students who have become interested enough to learn the programming language. In the process of doing exercises a student can modify an existing function and, although he can use it, he is unable to store the modified program into this system. This insures that the stored programs are protected and a student using the system after a user has made modifications for his own work will still be able to rely on the descriptions and instructions in the student manual.

The utilization of this interactive system is limited only by the size of the computer facility and the imagination of the user. The system is easy to use, data input is minimal, program listings show exactly what is happening in familiar notation, and the system can be modified and added to by any programmer familiar with the language.

CHAPTER IV

SYSTEM ENVIRONMENT

The computer hardware configuration of the Biometry Department at Medical College of Virginia - Virginia Commonwealth University on which the actual system programming was written includes the following [8]:

1. IBM 1130 CENTRAL PROCESSING UNIT(CPU) - Model 2C [16K]

   This model has a storage-access time of 3.6 microseconds. A single 16-bit word (two bytes) is accessed during each storage-access cycle. This model also has a single disk-storage drive housed in the CPU enclosure.

2. SINGLE DISK STORAGE - Model 2315

   Single disk storage is an auxiliary storage device housed within the 1131 CPU enclosure. It consists of a single disk drive and a removable disk cartridge. Storage capacity of each cartridge is 512,000 sixteen-bit words. The data transfer rate of the disk storage is 720,000 bits per second.

3. IBM 1442 CARD READ PUNCH - Model 6

   The Card Read Punch provides serial reading and punching of cards at the following rates:

   Read - 300 cards per minute

   Punch - 80 columns per second

4. IBM 1132 PRINTER*

   The 1132 Printer prints alphanumeric data at the following rates:

*The line printer is not used by APL/1130. Output is generated on the Central Processing Unit typewriter console.

Alphameric data - 80 lines per minute

Numeric data - 110 lines per minute

The computer language used in this project was APL/1130 which is a type III (contributed) program distributed, but not developed by International Business Machines Corporation. It is a disk oriented language and currently is designed to execute on any IBM 1130 machine with at least 8,000 words of core storage. The APL language is designed to give the user maximum flexibility in the execution of programs and in the general use of the computer. It has the additional capability of executing interactive statements one at a time, or it can be programmed with functions which can be stored and executed at a later time.

As soon as a user has signed onto the APL/1130 system a block of the internal storage is put at his disposal. This block of storage is called a Workspace (WS). Within this workspace all his functions and variables will be stored. This includes any temporary intermediate results while the calculations are in process. These workspaces are in turn divided into two parts: an F space (roughly 8,000 words) which is disk resident and contains the source statements of all defined functions of the given workspace; and an M space (1,920 words) which contains the variables, execution stack and the significant pointers and is core resident. The disk organization is such that 40 workspaces plus the APL System can be resident on a single 2315 cartridge. This is done by allocating workspaces on a track basis.

It is possible for a single user to have several different workspaces within the computer. However, only one of these is ever available for calculation at any one time. The one workspace which is currently available is called the active workspace.

The APL/1130 system has two modes of operation, called execution mode and definition mode. The computer is automatically in execution mode unless the user has specified differently. While in this mode the user enters a single line algebraic statement and the result is calculated immediately. The results will be printed and the computer will space down one line, indent and the keyboard will unlock allowing another expression to be entered. When the computer is in definition mode, it does not execute the instruction that is entered, but stores it as part of the definition of the program. The instructions that make up the program are not executed until (at some later time, when the system has returned to execution mode) a call is made for the execution of this program. It was the definition mode that was used in the actual programming of the Statistical System.

The APL Statistical System is contained on one of the forty-workspace disk cartridges. It is divided into two areas: STATSYSTEM and STATLIBRARY. The STATSYSTEM consists of one eight thousand word workspace. This is the workspace which the student automatically clears and activates when he "signs on" according to the instructions in the User's Manual. Once the user has signed on he may enter his data and operate on it using any of the APL operators (those necessary for the operation of this system are discussed in the Manual). The student may also use any of the statistical programs by copying the appropriate workspace from the STATLIBRARY.

The STATLIBRARY contains the workspaces in which are stored the necessary statistical methods associated with each particular workspace. In addition to the statistical method each workspace contains a SHOWME program. When executed by typing the word "SHOWME", this program will give a description of the primary functions stored there and an example showing how to enter the data and execute the function. The name of each workspace is associated with the statistical method it contains.

The following is a list and description of the workspaces stored in the STATLIBRARY. A more detailed explanation of each is to be found in the User's Manual under the SHOWME functions.

DESCRIPTION - contains a function which will give the mean, median, mode, variance, standard deviation, maximum, minimum and range of a sample of any size. In addition, if requested, the sample will be printed in ascending order.

PAIREDTTEST, EQUALVAR:TTEST and UNEQUALVAR:TTEST - These three workspaces each contain a function which will calculate the test statistic for one of the t-tests giving also the degrees of freedom associated with it. The EQUAL and UNEQUAL VARiance pertains to the assumptions of the t-test. For UNEQUALVAR the adjusted degrees of freedom associated with the statistic are given. The user also has the option of obtaining the confidence interval for the difference between the means.

MULTIRANGE - contains a function which calculates the test statistic for Duncan's Multiple Range test for equal or unequal sample sizes. The user must input the Mean Square Error and Least Significant Studentized Ranges.

BARTLETT - contains a function which computes Bartlett's test for homogeneity of variances. It will also perform three different transformations on the data: $\sqrt{X}$; $\text{Log}_e(X+1)$; and ARCSIN X and again perform Bartlett's test.

POLYNOMIAL - contains a function which will fit up to a fourth degree polynomial regression to a set of data. Output includes the vector of estimated regression coefficients and the total sum of squares partitioned into the sum of squares associated with each coefficient.

COVARIANCE - contains two functions. One function computes a one-way analysis of covariance with one covariate. The other, if requested by the user, calculates a Duncan's Multiple Range on the adjusted means. The sample sizes may be unequal.

MULTIREGR - contains a function which will fit a multiple regression. Output includes the calculations associated with the forward Doolittle method and the partitioned sum of squares.

ONEWAYCHISQ - contains a function which computes a χ^2 goodness of fit test statistic where the probabilities for each cell are equal or unequal, but specified.

TWOWAYCHISQ, THREEWAYCHI - contain functions which computes the test statistics for two and three way contingency tables.

REGRESSION - contains two functions. One fits a simple linear regression. The output includes the estimated regression equation; the correlation coefficient; a t-test for Beta equal to zero or some specified value; confidence limits for Beta, and a point on the true regression line. Prediction limits associated with the observed data point and predictions and residuals are also given. The second function automatically calculates the sum of squares associated with pure error and lack of fit if there are replicate observations.

EQUALLINES - contains a function which does the necessary calculations to test the following three hypothesis:

1. Equal intercepts

2. Equal slopes

3. 1 and 2 simultaneously

ONEWAYANOVA - contains a function which computes the necessary sums of squares and degrees of freedom for a one-way analysis of variance on equal or unequal sample sizes.

TWOWAYANOVA - contains a function which calculates a two-way analysis of variance with equal cell sizes.

UNEQUALSIZE:ANOVA2 - contains a function which calculates a two-way analysis of variance for unequal cell sizes.

INCOMPLETE:BLOCK - contains a function which calculates the appropriate sums of squares for a balanced incomplete block design.

GENERALREGR - contains a function which calculates the general regression signifiance test. Input includes an X matrix of full rank and a vector of observations.

Three additional workspaces are:

GENMETHODS - this workspace contains four specialized functions used in other workspace as subroutines, but are also stored here to be used seperately:

MEAN - calculates the estimated mean of a sample

VAR - calculates the estimated variance of a sample

JINV - takes the inverse of a matrix

DSORT - gives the subscripts of a sample which would reorder the sample in descending order.

DATAINPUT - contains a function which aids in inputing data into the active workspace.

LIBRARY - contains a SHOWME function which lists the names of the above workspaces and their associated statistical methods.

CHAPTER V

APL STATISTICAL SYSTEM USER'S MANUAL

INTRODUCTION

This manual is designed to give the limited instructions necessary to use the APL Statistical System. It introduces the basic system commands and operators used as they pertain to the Statistical System only. For a more detailed explanation of these operators the reader is referred to [2]. Specific examples are given throughout the manual with a broader example of an actual statistical problem illustrated at the end of the text. The SHOWME functions are displayed and following these is the appendix of program listings.

ORGANIZATION

The APL STATISTICAL SYSTEM is organized into two sections: the STATSYSTEM area which consists of one workspace and the STATLIBRARY which consists of a number of stored workspaces each containing a main function and needed subfunctions.

When a student "signs on" he activates the STATSYSTEM and is given a clear workspace. Into this active workspace he can copy workspaces from the STATLIBRARY. In each of these workspaces he will find a SHOWME function which when executed will display a description of the primary function in the workspace and an example. The student can then erase the SHOWME function providing more room in the active workspace for calculations and execute the program. He is also able to use any of the primitive operators (described in a later section) while in this active workspace.

STARTING THE SYSTEM

1. Turn on 1130, load APL STATISTICAL SYSTEM disk, and mount APL typing element.
2. Place APLIPL card followed by a blank card in card reader. Press start on the card reader.
3. When FILE READY light comes on, press IMMEDIATE STOP, RESET, and PROGRAM LOAD.
4. When the K B SELECT light comes on the user is ready to sign on.

SHIFTING, SIGNING ON AND SIGNING OFF

As the user sits at the 1130 keyboard he will notice in small white overlays the APL characters. On most keys he will see three characters per key. These are the lower, middle and upper shift characters.

1. Shifting

   In the upper left hand corner of the keyboard the user will find two keys with two white arrows each. These are the shift keys. The one on the left shifts from the usual lower case to upper and from middle to lower. The one on the right shifts from upper to lower and lower to middle.

   Pressing the left key shifts to upper case. Pressing it again locks the keyboard in upper case. The analogous is true of the right shift key.

2. Signing on

   a. Press the left shift key

   b. Type a right parenthesis,), found on the rightmost key of the second row of keys.

   c. Press the right shift key twice and type in the number 543 found across the second row of keys.

d. Press RETURN located just above the U key.

e. The console will respond with

STATSYSTEM SIGNED ON

A P L / 1 1 3 0

3. Signing off

a. Type)OFF

b. The console will respond with

SIGNED OFF

SYSTEM COMMANDS

For use with this system, the System Commands are divided into Workspace Control Commands and Inquiry Commands. The workspace control commands affect the state of the active workspace. They will be used as follows:

)CLEAR

Erases every function and variable in the active workspace.

)COPY 1130 followed by the full name of a workspace in the STATLIBRARY

A copy of the whole workspace will be copied into the active workspace. This is the command which is used to load into the active workspace one of the statistical methods from the program library.

)ERASE followed by the name of functions or variables

The named objects will be deleted.

The inquiry commands provide information without affecting the state of the system. They are used as follows:

)FNS

The names of the functions located in the active workspace will be listed.

)VARS

The names of all variables located in the active workspace will be listed.

)LIB 1130

The names of all the stored workspaces in the STATLIBRARY will be listed. In some instances passwords have been added to the workspace name to make the name more complete, e.g. EQUALVAR:TTEST. TTEST in this case is a password and would not be listed by the LIB command. For a complete listing of workspace names the workspace named LIBRARY should be copied and the SHOWME function displayed (See PROGRAM EXECUTION).

ERRORS

The following is a list of possible error messages the user might encounter, their causes and corrective actions necessary:

1. CHARACTER - Character overstrike
2. ID - Variable name with more than six characters.
3. LENGTH - Vectors not conformable for addition, subtraction, etc.
4. LINE TOO LONG - Data input line is longer than 160 characters.

   SOLUTION: Segment The Data (See Data Input)
5. RANK - Matrices are not conformable for addition, multiplication, etc.
6. SYNTAX - Invalid syntax, e.g., two variables juxtaposed; function used without appropriate arguments as dictated by its header (See Program Execution); mismatched parentheses.
7. VALUE - Use of a name has not been assigned a value.

   SOLUTION: Assign A Value To The Variable
8. WS FULL - Workspace is full   SOLUTION: Erase Needless Objects

If during the execution of a program, the program halts, types the name of the function and an error message, IMMEDIATELY type →0. Check your data to see that it is in the exact form described for that function or use one of the corrections listed above. Always be sure to exit from the program, by typing →0, before trying to correct the error.

DATA INPUT

There are two ways in which data can be entered in the system:

1. Keyboard entry of Data

   When a program is executed it is often the case that dummy arguments in the execution statement must be replaced by actual data or the name of some data variable previously defined. If the argument is a data array (matrix) it is recommended that a data variable be defined.

   To define a variable simply type the variable name, an arrow pointing toward that variable name, and the string of data, being sure to skip a space between each data point. If the data string is longer than the physical line across the page, strike the RETURN key. Continue by typing the variable name, an arrow pointing to it, the variable name again, a comma, and then the remainder of the data.

   For example,

   ```
   NAME← 4 16.3 ¯12 11 7 18 44 13 15.6 17.432 35 42 21 55 68
   NAME← NAME, 29 88 84 35 42 ¯86.2 93 48 85
   ```

   NOTE: The negative sign is different from the minus operator.
   The negative sign is found in the middle shift of the W key.

   After a data string is in the workspace it is often necessary to reshape the vector into a matrix. To accomplish this one uses the reshape operator, ρ. The ρ operator works by reshaping the variable to the right of it into the dimensions specified by the numbers to the left. For example,

```
NAME← 3 8 ρ NAME
```

NAME would now become equal to the matrix

4	16.3	¯12	11	7	18	44	13
15.6	17.432	35	42	21	55	68	29
88	84	35	42	¯86.2	93	48	85

If a mistake is made during the actual typing of the input, it is possible to correct the data by backspacing to the error and striking the ATTN (Attention) key located on the console keyboard. This will place a caret ∨ under the error and erase everthing in the line from that position on to the right. At this point the correct data point can be entered followed by the remaining elements.

If the data has already been entered a particular element can be replaced by using the subscript notation, [ ]. For example,

```
       S ← 2 2 ρ 3 4 5 6          (S becomes 2x2 matrix)

       S                          (display S)

3    4

5    6

       S[2;1] ← 7                 (replace the 2,1 element of
                                   S by 7)

       S                          (display S again)

3    4
                                  (5 has been replaced by 7)
7    6
```

The data can also be entered from the keyboard by using a special program called INPUT located in a workspace named DATAINPUT. To use this function type in the following:

1.) COPY 1130 DATAINPUT

2. NAME← D1 INPUT D2

where NAME is the variable name for the data

D1 is the number of rows of the matrix to be formed from the data

D1 = 1 for a vector

D2 is the number of columns in the matrix to be formed

This program will then keep asking for data by typing ⎕: until enough elements have been entered to form the specified array. If one finds that there is not enough data to satisfy the dimensions he has specified, he must continue inputing numbers until the program exits normally and then start at step 2 again.

2. Card input of Data

To prepare data for card input punch on cards exactly what is shown for keyboard input with the following conditions:

a) Punch from card column 1 through card column 71. Do not punch anything beyond column 71.

b) Substitute # for ←; and @RHO for ρ on the IBM 029 keypunch machine.

c) When using card input for the INPUT program type in)CARD the first time ⎕: appears. (See Example)

d) A card with)CARD END punched on it and a BLANK CARD must follow the card input for the keyboard to unlock for the next entry.

During the execution of many of the STATSYSTEM programs, inquiries are made by the program. Usually the responses are entered through the keyboard though cards with the appropriate responses punched on them may be substituted. If such is the case, at the first inquiry)CARD should be typed in. From then until either the ATTN key on the keyboard is pressed or a)CARD END card is encountered the system will expect card input every time it would have received keyboard input.

DATA OUTPUT

1. Console typewriter

Output from each program is illustrated in the examples of the SHOWME functions. In addition to the programmed output, certain variables generated within each function can be displayed by entering the name of that variable. For instance, after execution of any of the t-test programs it is possible to type NUM or DEN and display the numerator or denominator of the t-statistics. To determine other available program generated variables see the appropriate SHOWME output.

In general, if data or any operation is given a variable name, the results of that operator can be displayed by typing the name of that variable. For example,

```
X←3 8 4 6   (X becomes 3 8 4 6)
X           (display X)
3 8 4 6
```

It should be noted that any time the computer returns any results it will start at the left hand margin. If the keyboard is unlocked for an entry the typing element will indent seven spaces.

2. Cards

To have output punched onto cards it is necessary first to give the data a variable name. Place blank cards in the card reader and press start. Type)PCH followed by the name or names of objects to be punched and a copy of all data defined by those names will be punched onto cards in free field format skipping a space between each data point.

PRIMITIVE OPERATORS

Only four of the APL operators needed to complete some of the programmed calculations in the system are discussed. They are + - × ÷ and are used to calculate, for example, the F-ratio after the Sum of Squares are calculated in the Analysis of Variance programs.

Examples follow for each operator:

2+3	(2 plus 3)
5	(result)
18-6	(18 minus 6)
12	(result)
4×6	(4 times 6)
24	(result)
(14÷3)÷(17÷8)	(result of 14 divided by 3 divided by result of 17 divided by 8)
2.1957	(result)
13-¯4	(13 minus a negative 4)
17	(result)
¯1×16	(negative 1 times 16)
¯16	(result)

PROGRAM EXECUTION

To execute a program which is in the active workspace the user simply types in the program header. This program header consists of the program name and one or two dummy arguments which must be replaced by either actual data or a data variable name. If the header has one dummy variable it will appear to the right of the program name. If there are two dummy variables they will appear on each side of the program name. Each of these function headers is explained in detail in each SHOWME function (executed by typing SHOWME) associated with that particular workspace in which the program is stored.

Many of these programs will make inquiries for input during the execution of the program. Each inquiry should be self explanatory and the user will either respond with data, a data variable name, or a literal response. If the inquiry is for literal input the inquiry will include the choices of response in quotes. The user should choose one of these keywords as his response. For other than literal input the inquiry will end with ⎕: indicating data of a data variable name is required.

FUNCTION DISPLAY

After a function has been copied into the active workspace it is possible to display the complete program by typing ∇NAME [⎕]∇ where NAME is the name of the function. When displaying a function it may be helpful to refer to page 197 of the APL Primer [2].

EXAMPLE

Consider a set of data in which each column consists of the number of flies, taken at a particular time of day, found on a fixed number of dairy cattle. It is important to know whether there is a significant difference among the numbers of flies found at specific times during the day.

These data will be analyzed by performing an Analysis of Variance. However, before such an analysis can be accomplished it is necessary to test for the equality of variances. If rejected then the data should be transformed and a Bartlett's test performed on the transformed data. Once we have equal variances the Analysis of Variance can be run. If significant then a Duncan's multiple range test should be performed.

The following pages are a reproduction of the actual entries and responses which would take place in performing the analysis on the APL Statistical System.

```
      )543
STATSYSTEM SIGNED ON

                   A P L \ 1 1 3 0

      )COPY 1130 DATAINPUT
      PROB←12 INPUT 5
⎕:
      )CARD
⎕:
      18 13 18 42 17 5 7 25 17 75 5 5 9 32 23 5 17 8 24
⎕:
      21 2 4 10 14 24 3 5 14 23 19 5 10 5 11 43 2 2 20
⎕:
      30 25 4 9 11 9 35 5 16 7 11 34 2 3 10 16 17 4 5
⎕:
      7 11 27
      )CARD END
      PROB
     .
   18   13   18   42   17
    5    7   25   17   75
    5    5    9   32   23
    5   17    8   24   21
    2    4   10   14   24
    3    5   14   23   19
    5   10    5   11   43
    2    2   20   30   25
    4    9   11    9   35
    5   16    7   11   34
    2    3   10   16   17
    4    5    7   11   27
      )COPY 1130 BARTLETT
      BARTLT

ENTER NUMBER OF CELLS:
⎕:
      5
ENTER DATA FROM CELL 1:
⎕:
      PROB[;1]
ENTER DATA FROM CELL 2:
⎕:
      PROB[;2]
ENTER DATA FROM CELL 3:
⎕:
      PROB[;3]
ENTER DATA FROM CELL 4:
⎕:
      PROB[;4]
ENTER DATA FROM CELL 5:
⎕:
```

```
      PROB[;5]

BARTLETT'S STATISTIC = 26.6119 WITH 4 DEGREES OF FREEDOM.

DO YOU WISH TO TRY SQUARE 'ROOT', 'LOG BASE E', 'ARCSIN'
TRANSFORMATION OR 'NO' TRANSFORMATION?
LOG BASE E

BARTLETT'S STATISTIC = 1.313 WITH 4 DEGREES OF FREEDOM.

DO YOU WISH TO TRY SQUARE 'ROOT', 'LOG BASE E', 'ARCSIN'
TRANSFORMATION OR 'NO' TRANSFORMATION?
NO

      PROB←⍟PROB+1
      PROB

 2.94444      2.63906      2.94444      3.7612       2.89037
 1.79176      2.07944      3.2581       2.89037      4.33073
 1.79176      1.79176      2.30259      3.49651      3.17805
 1.79176      2.89037      2.19722      3.21888      3.09104
 1.09861      1.60944      2.3979       2.70805      3.21888
 1.38629      1.79176      2.70805      3.17805      2.99573
 1.79176      2.3979       1.79176      2.48491      3.78419
 1.09861      1.09861      3.04452      3.43399      3.2581
 1.60944      2.30259      2.48491      2.30259      3.58352
 1.79176      2.83321      2.07944      2.48491      3.55535
 1.09861      1.38629      2.3979       2.83321      2.89037
 1.60944      1.79176      2.07944      2.48491      3.3322
      )COPY 1130 ONEWAYANOVA
      5 ANOVA1 PROB
ANOVA TABLE
SUM OF SQUARES TOTAL = 34.7608 WITH 59 DEGREES OF FREEDOM.
SUM OF SQUARES TREATMENTS = 21.9221 WITH 4 DEGREES OF FREEDOM.
SUM OF SQUARES ERROR = 12.8387 WITH 55 DEGREES OF FREEDOM.
DO YOU WISH TO RUN SINGLE DEGREE OF FREEDOM 'CONTRASTS'
OR 'NO' CONTRASTS?
NO
      (21.9221÷4)÷(12.8387÷55)
23.4781
      )COPY 1130 GENMETHODS MEAN VAR
      MEAN PROB
1.65035  2.05102  2.47386  2.9398  3.34238
      VAR PROB
0.250997  0.326557  0.191899  0.223439  0.174269
      )COPY 1130 MULTIRANGE
      12 12 12 12 12 DUNCAN MEAN PROB

ENTER M.S. ERROR:
⎕:
      12.8387÷55
ENTER APPROPRIATE VALUES FOR LEAST SIGNIFICANT STUDENTIZED RANGE:
⎕:
```

```
      2.83 2.98 3.08 3.14

MEANS:
 3.34238       2.9398       2.47386      2.05102      1.65035
 5             4            3            2            1

3.34238 IS SIGNIFICANTLY DIFFERENT FROM 2.9398  2.47386  2.05102
  1.65035
2.9398 IS SIGNIFICANTLY DIFFERENT FROM 2.47386  2.05102  1.65035
2.47386 IS SIGNIFICANTLY DIFFERENT FROM 2.05102  1.65035
2.05102 IS SIGNIFICANTLY DIFFERENT FROM 1.65035

      )OFF
SIGNED OFF
```

SHOWME

The following pages are the displayed SHOWME's stored in each of the APL Statistical System workspaces.

```
)COPY 1130 LIBRARY
SHOWME
```

WORKSPACE NAME:	ASSOCIATED METHOD:
DESCRIPTION	DESCRIPTIVE STATISTICS
EQUALVAR:TTEST	T-TEST (ASSUMING EQUAL VARIANCES)
UNEQUALVAR:TTEST	T-TEST (ASSUMING UNEQUAL VARIANCES)
PAIREDTTEST	PAIRED T-TEST
BARTLETT	BARTLETT'S TEST FOR EQUAL VARIANCES
MULTIRANGE	DUNCAN'S MULTIPLE RANGE TEST
ONEWAYANOVA	ONE-WAY ANALYSIS OF VARIANCE
TWOWAYANOVA	TWO-WAY ANALYSIS OF VARIANCE
REGRESSION	SIMPLE LINEAR REGRESSION
MULTIREGR	MULTIPLE LINEAR REGRESSION
POLYNOMIAL	ORTHOGONAL POLYNOMIAL REGRESSION
EQUALLINES	TESTS EQUALITY OF SLOPES AND INTERCEPTS
ONEWAYCHISQ	ONE-WAY CHI-SQUARE
TWOWAYCHISQ	TWO-WAY CHI-SQUARE
THREEWAYCHI	THREE-WAY CHI-SQUARE
COVARIANCE	ANALYSIS OF COVARIANCE
UNEQUALSIZE:ANOVA2	TWO-WAY ANOVA (UNEQUAL SIZE CELLS)
BALINCOMPLT:BLOCK	ANOVA FOR BALANCED INCOMPLETE BLOCKS
GENERALREGR	GENERAL REGRESSION SIGNIFICANCE TEST
DATAINPUT	DATA INPUT ROUTINE
GENMETHODS	UTILITY FUNCTIONS (NO SHOWME)
LIBRARY	LIST OF WORKSPACES

```
      )COPY 1130 DESCRIPTION
      SHOWME

FUNCTION: LOOK X                WORKSPACE: DESCRIPTION
   X = DATA VECTOR


THIS FUNCTION CALCULATES THE MEAN, MEDIAN, MODE, VARIANCE,
STANDARD DEVIATION, MAXIMUM, MINIMUM, AND RANGE OF ANY SIZE
SAMPLE, X.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

LET OUR SAMPLE, X ← 25  42  33  16  11  19  42  13  6  5  12
NOW TO EXECUTE THE PROGRAM WE TYPE:

LOOK X

MEAN = 20.3636
MEDIAN = 16
MODE = 42
VARIANCE = 179.254
STANDARD DEVIATION = 13.3886
LARGEST VALUE = 42
SMALLEST VALUE = 5
RANGE = 37


IF YOU WISH TO SEE THE SAMPLE REORDERED IN
ASCENDING ORDER SIMPLY TYPE ORD...

ORD
5  6  11  12  13  16  19  25  33  42  42
```

```
      )COPY 1130 EQUALVAR:TTEST
      SHOWME

FUNCTION: X1 TTEST1 X2         WORKSPACE: EQUALVAR:TTEST
  X1 = FIRST SAMPLE
  X2 = SECOND SAMPLE

THIS FUNCTION CALCULATES STUDENT'S T-STATISTIC WHEN VAR-
IANCES ARE UNKNOWN BUT ASSUMED EQUAL.  OPTIONAL OUTPUT IS
THE CONFIDENCE INTERVAL FOR MU[1]-MU[2].

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR TTEST1
 SOURCE: STEEL AND TORRIE, PRINCIPLES AND PROCEDURES OF
         STATISTICS, PAGE 74.

OUR FIRST SAMPLE:
A ← 57.8  56.2  61.9  54.4  53.6  56.4  53.2
OUR SECOND SAMPLE:
B ← 64.2  58.7  63.1  62.5  59.8  59.2

IF WE WISH TO SEE THE CONFIDENCE INTERVAL FOR THE DIFFERENCES
OF THE MEANS, WE RESPOND POSITIVELY TO THE INQUIRY AND WHEN
ASKED TO TYPE IN THE APPROPRIATE T-VALUE, WE TYPE IN THE TABLE
VALUE FOR THE APPROPRIATE DEGREES OF FREEDOM.  IN OUR CASE THE
T-VALUE WOULD BE 2.201.

TO EXECUTE THE PROGRAM WE TYPE:

A TTEST1 B

T = 3.34416
DF = 11

DO YOU WANT THE CONFIDENCE 'INTERVAL' FOR MU1-MU2 OR 'NO' INTERVAL?
INTERVAL
ENTER T-VALUE:
⎕:
      2.201
1.7214 <MU1-MU2< 8.35003


IF YOU WISH TO SEE THE NUMERATOR OR DENOMINATOR OF THE
TEST STATISTIC, TYPE NUM OR DEN RESPECTIVELY...
NUM
5.03571

DEN
1.50582
```

```
      )COPY 1130 UNEQUALVAR:TTEST
      SHOWME

FUNCTION: X1 TTEST2 X2        WORKSPACE: UNEQUALVAR:TTEST
  X1 = FIRST SAMPLE
  X2 = SECOND SAMPLE

THIS FUNCTION CALCULATES THE STUDENT'S T-STATISTIC WHEN
THE VARIANCES ARE UNKNOWN AND ASSUMED TO BE UNEQUAL. OUTPUT
INCLUDES THE ADJUSTED DEGREES OF FREEDOM. OPTIONAL OUTPUT
IS THE CONFIDENCE INTERVAL FOR MU[1]-MU[2].

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR TTEST2
 SOURCE: LI, INTRODUCTION TO EXPERIMENTAL STATISTICS,
         PAGE 434.

OUR FIRST SAMPLE, A =
11  18  15  26  25  22  11  13  19  10
OUR SECOND SAMPLE, B =
26  21  19  22  19  27  28  22  24  22
TO EXECUTE THE PROGRAM WE TYPE:

A TTEST2 B

T = 2.82494
THE APPROXIMATE DEGREES OF FREEDOM EQUAL 14

DO YOU WANT THE CONFIDENCE 'INTERVAL' FOR MU1-MU2 OR
 'NO' INTERVAL?
NO
```

```
      )COPY 1130 PAIREDTTEST
      SHOWME

FUNCTION: X1 PTTEST X2        WORKSPACE: PAIREDTTEST
  X1 = FIRST SAMPLE
  X2 = SECOND SAMPLE

THIS FUNCTION CALCUALES PAIRED T-TEST STATISTIC.  OPTIONAL
OUTPUT IS CONFIDENCE INTERVAL FOR MU[1]-MU[2]. SAMPLE SIZES
MUST BE EQUAL.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR PTTEST
 SOURCE: STEEL AND TORRIE, PRINCIPLES AND PROCEDURES OF
         STATISTICS, PAGE 79.

OUR FIRST SAMPLE:
A ← 62.5  65.2  67.6  69.9  69.4  70.1  67.8  67  68.5  62.4
OUR SECOND SAMPLE:
B ← 51.7  54.2  53.3  57  56.4  61.5  57.2  56.2  58.4  55.8
IF WE WISH TO SEE THE CONFIDENCE INTERVAL FOR THE DIFFERENCES
OF THE MEANS, WE RESPOND POSITIVELY TO THE INQUIRY AND WHEN
ASKED TO TYPE IN THE APPROPRIATE T-VALUE, WE TYPE IN THE TABLE
VALUE FOR THE APPROPRIATE DEGREES OF FREEDOM.  IN OUR CASE THE
T-VALUE WOULD BE 3.301.

NOW TO EXECUTE THE PROGRAM WE TYPE:

A PTTEST B

T = 15.4585
DF = 9

DO YOU WANT THE CONFIDENCE 'INTERVAL' FOR MU1-MU2 OR 'NO' INTERVAL?
INTERVAL
ENTER T-VALUE:
⎕:
      3.301
8.54882 <MU1-MU2< 13.1912



IF YOU WISH TO SEE THE NUMERATOR OR DENOMINATOR OF THE
TEST STATISTIC, TYPE NUM OR DEN RESPECTIVELY...

NUM
10.87

DEN
0.703175
```

```
      )COPY 1130 BARTLETT
      SHOWME

FUNCTION:BARTLT                    WORKSPACE: BARTLETT

THIS FUNCTION CALCULATES THE TEST STATISTIC FOR BARTLETT'S
TEST OF EQUALITY OF VARIANCES. THE USER HAS THE OPTION OF
CALLING ANY (OR ALL) OF THE FOLLOWING THREE DATA TRANSFORMATIONS:
SQUARE ROOT, NATURAL LOG (X+1), AND ARCSIN.  THE SAMPLES MAY BE
UNEQUAL.
AFTER THE APPROPRIATE TRANSFORMATION HAS BEEN FOUND THE DATA
MAY BE TRANSFORMED BY THE FOLLOWING:
      FOR SQUARE ROOT:  NAME←NAME*.5
      FOR NATURAL LOG(X+1): NAME←⍟NAME+1
      FOR ARCSIN: NAME←ARCSIN NAME
     WHERE NAME IS THE VARIABLE NAME FOR THE DATA TO BE TRANS-
     FORMED.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR BARTLT
 SOURCE: LI, INTRODUCTION TO EXPERIMENTAL STATISTICS,
         PAGE 439.

SAMPLE FOR GROUP 1, A ← 6  9  3
SAMPLE FOR GROUP 2, B ← 10  4  9  12  8  11
SAMPLE FOR GROUP 3, C ← 2  6  4

WE WILL NOW EXECUTE THE PROGRAM.  AS INQUIRIES ARE MADE
TYPE IN THE APPROPRIATE RESPONSE...
FOR INSTANCE, WHEN DATA FROM CELL 1 IS ASKED FOR TYPE:
6 9 3
OR SIMPLY TYPE IN THE APPROPRIATE VARIABLE NAME, A.

NOW TO EXECUTE THE PROGRAM WE TYPE...

BARTLT

ENTER NUMBER OF CELLS:
⎕:
      3
ENTER DATA FROM CELL 1:
⎕:
      A
ENTER DATA FROM CELL 2:
⎕:
      B
ENTER DATA FROM CELL 3:
⎕:
      C

BARTLETT'S STATISTIC = 0.31116 WITH 2 DEGREES OF FREEDOM.
```

```
DO YOU WISH TO TRY SQUARE 'ROOT', 'LOG BASE E', 'ARCSIN'
TRANSFORMATION OR 'NO' TRANSFORMATION?
LOG BASE E

BARTLETT'S STATISTIC = 0.260408 WITH 2 DEGREES OF FREEDOM.

DO YOU WISH TO TRY SQUARE 'ROOT', 'LOG BASE E', 'ARCSIN'
TRANSFORMATION OR 'NO' TRANSFORMATION?
NO

WE WILL NOW DO THE ACTUAL NATURAL LOG (X+1) TRANSFORMATION
ON THE DATA...
A←⍟A+1
B←⍟B+1
C←⍟C+1

A
1.94591  2.30259  1.38629
B
2.3979  1.60944  2.30259  2.56495  2.19722  2.48491
C
1.09861  1.94591  1.60944
```

```
      )COPY 1130 MULTIRANGE
      SHOWME

FUNCTION: K DUNCAN MEANS     WORKSPACE: MULTIRANGE
   K = A VECTOR INDICATING THE SAMPLE SIZE FOR EACH MEAN.
       K(1) WOULD BE THE SAMPLE SIZE FOR THE FIRST MEAN...
   MEANS = VECTOR OF MEANS ON WHICH THE TEST WILL BE
           CALCULATED.

THIS FUNCTION CALCULATES DUNCAN'S MULTIPLE RANGE TEST ON
ANY NUMBER OF MEANS WITH EQUAL OR UNEQUAL SAMPLE SIZES.
THIS FUNCTION SHOULD BE EXECUTED AFTER ONE OF THE ANALYSIS
OF VARIANCE FUNCTIONS AS MEAN SQUARE (ERROR) IS NEEDED FOR
'DUNCAN'. THE USER MUST ALSO ENTER THE APPROPRIATE VALUES
FOR THE LEAST SIGNIFICANT STUDENTIZED RANGES FROM ONE OF THE
AVAILABLE TABLES.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR DUNCAN
 SOURCE: BIOMETRICS:12 PAGE 307

THE MEANS, X ← 521  528  564  498  630  458
THE SAMPLE SIZES, S ← 4  3  5  5  2  3
THE MEAN SQUARE ERROR, MSE ← 2397
THE STUDENTIZED RANGE FOR 16 DEGREES OF FREEDOM,
SR ← 3  3.149  3.23  3.3  3.34

NOW WE WILL EXECUTE THE PROGRAM.  AS INQUIRIES ARE MADE
TYPE IN THE VARIABLE NAME OF EACH RESPONSE...
FOR INSTANCE WHEN THE MEAN SQUARE ERROR IS ASKED FOR TYPE
IN MSE, THE NAME FOR THAT VARIABLE.

S DUNCAN X

ENTER M.S. ERROR:
⎕:
      MSE
ENTER APPROPRIATE VALUES FOR LEAST SIGNIFICANT STUDENTIZED RANGE:
⎕:
      SR

MEANS:
     630      564      528      521      498      458
       5        3        2        1        4        6

630 IS SIGNIFICANTLY DIFFERENT FROM 528  521  498  458
564 IS SIGNIFICANTLY DIFFERENT FROM 458
```

```
      )COPY 1130 ONEWAYANOVA
      SHOWME

FUNCTION: K ANOVA1 X          WORKSPACE: ONEWAYANOVA
  K = NUMBER OF TREATMENT LEVELS
  X = (M,K) DATA MATRIX
         M = NUMBER OF OBSERVATIONS
     IF X EQUALS ANY SCALAR THE DATA WILL BE ASKED FOR A
     TREATMENT AT A TIME (USED FOR UNEQUAL SAMPLE SIZES)

THIS FUNCTION CALCULATES THE NECESSARY SUMS OF SQUARES FOR ONE
WAY ANALYSIS OF VARIANCE WITH EQUAL OR UNEQUAL SAMPLE SIZES.
USER HAS THE OPTION OF ENTERING CONTRAST COEFFICIENTS FOR
SUM OF SQUARES CONTRAST.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR ANOVA1
 SOURCE: FRYER, CONCEPTS AND METHODS OF EXPERIMENTAL
         STATISTICS, PAGE 272.

DATA FOR GROUP 1:
A ← 0.785  0.55  0.345  0.5  0.66  0.79  0.81
DATA FOR GROUP 2:
B ← 0.875  0.9  0.655  0.725  0.845  0.6  0.78  0.82
DATA FOR GROUP 3:
C ← 0.61  0.775  0.895  0.75  0.82
DATA FOR GROUP 4:
D ← 0.905  0.68  0.77  0.885  0.805  0.59

SINCE WE HAVE UNEQUAL REPLICATIONS WE SHOULD MAKE X A SCALAR
AND INPUT THE DATA A TREATMENT AT A TIME.

TO EXECUTE THE PROGRAM WE TYPE:

4 ANOVA1 99
INPUT DATA FOR TREATMENT 1...
⎕:
      A
INPUT DATA FOR TREATMENT 2...
⎕:
      B
INPUT DATA FOR TREATMENT 3...
⎕:
      C
INPUT DATA FOR TREATMENT 4...
⎕:
      D
```

```
ANOVA TABLE
SUM OF SQUARES TOTAL = 0.483063 WITH 25 DEGREES OF FREEDOM.
SUM OF SQUARES TREATMENTS = 0.0983524 WITH 3 DEGREES OF FREEDOM.
SUM OF SQUARES ERROR = 0.38471 WITH 22 DEGREES OF FREEDOM.
DO YOU WISH TO RUN SINGLE DEGREE OF FREEDOM 'CONTRASTS' OR
 'NO' CONTRASTS?
CONTRASTS
ENTER NUMBER OF CONTRASTS:
⎕:
      1
ENTER CONTRAST COEFFICIENT SET NUMBER 1...
⎕:
      ¯1 0 1 0
SUM OF SQUARES FOR CONTRAST 1 = 0.0290083
```

```
      )COPY 1130 TWOWAYANOVA
      SHOWME

FUNCTION: ANOVA2 X            WORKSPACE: TWOWAYANOVA
  X = (M×N) DATA MATRIX
        M = B TIMES R
        B = LEVELS OF ROW TREATMENT
        R = NUMBER OF REPLICATIONS IN EACH CELL
        N = LEVELS OF COLUMN TREATMENT

THIS FUNCTION CALCULATES THE SUMS OF SQUARES NECESSARY FOR A
TWO WAY ANALYSIS OF VARIANCE WITH EQUAL REPLICATIONS IN EACH
CELL. THE USER HAS THE OPTION OF ENTERING CONTRAST COEFFICIENTS
FOR SS(CONTRAST) FOR EITHER MAIN EFFECT OR FOR THE INTERACTION
TERM.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR ANOVA2
CONSIDER THE FOLLOWING:

OUR DATA IS:          TREATMENT A
                    1     2     3
                   [4    [2    [5       CELLS ARE BETWEEN
                 1  7     3     6       BRACKETS
                    5]    2]    4]
 TREATMENT B
                   [9    [8   [10
                 2  8     7     8
                    8]    5]    7]
IN THIS EXAMPLE X =

    4    2    5
    7    3    6
    5    2    4
    9    8   10
    8    7    8
    8    5    7

TO EXECUTE WE TYPE:

ANOVA2 X

ENTER NUMBER OF REPLICATIONS IN EACH CELL
⎕:
      3

ANOVA TABLE
SUM OF SQUARES TOTAL = 95.9994 WITH 17 DEGREES OF FREEDOM
SUM OF SQUARES TREATMENT A = 20.3328 WITH 2 DEGREES OF FREEDOM
SUM OF SQUARES TREATMENT B = 56.8881 WITH 1 DEGREES OF FREEDOM
SUM OF SQUARES INTERACTION = 1.44531 WITH 2 DEGREES OF FREEDOM
SUM OF SQUARES ERROR = 17.3333 WITH 12 DEGREES OF FREEDOM

DO YOU WISH TO DO CONTRASTS ON 'COLUMNS','ROWS', 'INTERACTION'
```

```
OR 'NO' CONTRASTS?
COLUMNS
ENTER CONTRAST COEFFICIENTS FOR TREATMENT A
⎕:
      1 0 ¯1
SUM OF SQUARES CONTRAST = 0.0833333

DO YOU WISH TO DO CONTRASTS ON 'COLUMNS','ROWS', 'INTERACTION'
OR 'NO' CONTRASTS?
ROWS
ENTER CONTRAST COEFFICIENTS FOR TREATMENT B
⎕:
      1 ¯1
SUM OF SQUARES CONTRAST = 56.8889

DO YOU WISH TO DO CONTRASTS ON 'COLUMNS','ROWS', 'INTERACTION'
OR 'NO' CONTRASTS?
INTERACTION
ENTER CONTRAST COEFFICIENTS FOR TREATMENT A
⎕:
      1 0 ¯1
ENTER CONTRAST COEFFICIENTS FOR TREATMENT B
⎕:
      1 ¯1
SUM OF SQUARES CONTRAST = 0.0833333

DO YOU WISH TO DO CONTRASTS ON 'COLUMNS','ROWS', 'INTERACTION'
OR 'NO' CONTRASTS?
NO
```

```
      )COPY 1130 REGRESSION
      SHOWME

FUNCTION: X REGR Y            WORKSPACE: REGRESSION
  X = VECTOR OF INDEPENDENT OBSERVATIONS
  Y = VECTOR OF DEPENDENT OBSERVATIONS

THIS FUNCTION COMPUTES THE PRODUCT MOMENT CORRELATION COEFFI-
CIENT AND THE REGRESSION EQUATION.  AS OPTIONAL OUTPUT THE FOL-
LOWING WILL BE GIVEN:
  T-TEST FOR BETA EQUAL TO ZERO OR SOME SPECIFIED VALUE
  CONFIDENCE INTERVAL FOR BETA
  CONFIDENCE INTERVAL ON A SPECIFIED POINT ON THE REGRESSION
    LINE
  PREDICTION LIMITS FOR A FUTURE OBSERVATION AT ANY X VALUE
  PREDICTED VALUES AND RESIDUALS
IN ADDITION, WHENEVER REPEATED MEASURES ARE ENCOUNTERED IN A
SET OF DATA, SS(PURE ERROR) AND SS(LACK OF FIT) ARE COMPUTED.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

CONSIDER THE FOLLOWING DATA
X ← 1  4  4  6  6  6
Y ← 7  5  3  3  2  1

WE SHOULD TEST BETA EQUAL TO ZERO.
WE WILL ALSO WANT TO LOOK AT THE CONFIDENCE INTERVALS WITH
T = 2.776 AND X = 2.

TO EXECUTE THE PROGRAM WE TYPE:

X REGR Y

REGRESSION EQUATION:  Y=3.5+¯1(X-4.5)
CORRELATION COEFFICIENT:¯0.910927

DO YOU WISH TO TEST BETA EQUAL TO 'BETA NOUGHT' OR 'ZERO' OR
'NO' TEST?
ZERO
T-VALUE = ¯4.41588 WITH 4 DEGREES OF FREEDOM .

SUM OF SQUARES FOR PURE ERROR = 4 WITH 3 DEGREES OF FREEDOM.
SUM OF SQUARES LACK OF FIT = 0 WITH 1 DEGREES OF FREEDOM:

DO YOU WISH CONFIDENCE 'LIMITS' OR 'NO' LIMITS?
LIMITS
INPUT T-VALUE AND VALUE FOR X:
⎕:
      2.776 2

CONFIDENCE INTERVALS
1- ON BETA
     U.L. = ¯0.37136
     L.L. = ¯1.62864
```

2- *ON A POINT ON REGRESSION LINE (X= 2)*
U.L. = 7.9376
L.L. = 4.0624
3- *PREDICTION LIMITS (X= 2)*
U.L. = 8.03703
L.L. = 3.96297

DO YOU WISH TO HAVE PREDICTIONS AND 'RESIDUALS' PRINTED OR 'NOT'?
RESIDUALS

X, Y, Y-PREDICTED, AND RESIDUALS:

1	7	7	0
4	5	4	1
4	3	4	-1
6	3	2	1
6	2	2	0
6	1	2	-1

```
      )COPY 1130 MULTIREGR
      SHOWME

FUNCTION: X MREG Y              WORKSPACE: MULTIREGR
  X = (P×N) MATRIX WHERE P = NUMBER OF INDEPENDENT VARIABLES
                         N = NUMBER OF OBSERVATIONS
  Y =  VECTOR OF OBSERVATIONS

THIS FUNCTION CALCULATES A MULTIPLE REGRESSION USING THE
DOOLITTLE METHOD.  OUTPUT INCLUDES THE RESULTS OF THE FORWARD
DOOLITTLE METHOD, SUM OF SQUARES TOTAL PARTITIONED INTO ONE
DEGREE OF FREEDOM SUM OF SQUARES ATTRIBUTED TO EACH COEFFICIENT
ADJUSTED FOR THE PRECEDING VARIABLE.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR MREG
 SOURCE: STEEL AND TORRIE, PRINCIPLES AND PROCEDURES OF
         STATISTICS, PAGES 282-296.

THE DATA FOR THIS EXAMPLE IS FOUND ON PAGE 282 OF STEEL AND
TORRIE AND WILL NOT BE LISTED HERE BECAUSE OF LACK OF SPACE.
THE X MATRIX HAS BEEN CORRECTED FOR THE MEAN AND HAS DIMEN-
SIONS EQUAL TO 30×3, HAVING 30 OBSERVATIONS FOR EACH OF THREE
VARIABLES.
THE DEPENDENT VARIABLE, Y, IS A VECTOR OF LENGTH 30.

TO EXECUTE THE PROGRAM WE TYPE:

X MREG Y
(FORWARD DOOLITTLE METHOD)
 9.84555       2.14131       1.67054      ¯5.82475       7.83264
 2.14131       10.6209       7.63666      ¯4.21148       16.1874
 1.67054       7.63666       33.0829       2.66835       45.0585


 9.84555       2.14131       1.67054      ¯5.82475       7.83264
 1             0.21749       0.169675     ¯0.591613      0.795552
 0             10.1552       7.27333      ¯2.94465       14.4839
 0             1             0.716216     ¯0.289964      1.42625
 0             0             27.5902       5.76568       33.3558
 0             0             1             0.208976      1.20898

ANOVA TABLE
SUM OF SQUARES REGRESSION = 5.50473 WITH 3 DEGREES OF FREEDOM.
   SUM OF SQUARES X[1] = 3.446 WITH 1 DEGREE OF FREEDOM.
   SUM OF SQUARES X[2]|X[1] = 0.853843 WITH 1 DEGREE OF FREEDOM.
   SUM OF SQUARES X[3]|X[1  2] = 1.20489 WITH 1 DEGREE OF FREEDOM.
SUM OF SQUARES RESIDUAL = 1.18479 WITH 26 DEGREES OF FREEDOM.
SUM OF SQUARES TOTAL = 6.68952 WITH 29 DEGREES OF FREEDOM.


BETA VECTOR EQUALS: 0.686  ¯0.531455  ¯0.439636  0.208976
```

```
      )COPY 1130 POLYNOMIAL
      SHOWME


FUNCTION: N POLYR DATA           WORKSPACE: POLYNOMIAL
  N = DEGREE OF POLYNOMIAL DESIRED  N≤4
  DATA = DATA MATRIX (M,2) WHERE M EQUALS NUMBER OF OBSERVA-
         TIONS (FIRST COLUMN IS INDEPENDENT VARIABLE AND
                SECOND COLUMN IS DEPENDENT VARIABLE)

THIS FUNCTION CALCULATES AN N DEGREE ORTHOGONAL POLYNOMIAL
REGRESSION. OUTPUT INCLUDES BETA VECTOR AND PARTITIONING OF
THE TOTAL SUM OF SQUARES INTO SINGLE DEGREE OF FREEDOM SUMS
OF SQUARES ASSOCIATED WITH EACH COEFFICIENT.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR POLYR
 SOURCE: KENDALL AND STUART, THE ADVANCED THEORY OF STATISTICS,
         VOL. 2, PAGE 360

LET X EQUAL
 1820         10.16
 1830         12
 1840         13.9
 1850         15.91
 1860         17.93
 1870         20.07
 1880         22.71
 1890         25.97
 1900         29
 1910         32.53
 1920         36.07
 1930         37.89
 1940         39.9

WE WOULD LIKE A FOURTH DEGREE POLYNOMIAL FIT...
TO EXECUTE WE WILL TYPE:

4 POLYR X

THE BETA VECTOR EQUALS:
24.1569  2.60698  0.060984  ¯0.0116348  ¯0.00325013

ANOVA TABLE
SUM OF SQUARES BETA[1] = 1236.93 WITH 1 DEGREE OF FREEDOM.
SUM OF SQUARES BETA[2] = 7.44554 WITH 1 DEGREE OF FREEDOM.
SUM OF SQUARES BETA[3] = 2.78751 WITH 1 DEGREE OF FREEDOM.
SUM OF SQUARES BETA[4] = 2.11306 WITH 1 DEGREE OF FREEDOM.
SUM OF SQUARES RESIDUAL = 0.427002 WITH 8 DEGREES OF FREEDOM.
SUM OF SQUARES TOTAL = 1249.71 WITH 12 DEGREES OF FREEDOM.
```

```
      )COPY 1130 EQUALLINES
      SHOWME

FUNCTION: LINES M              WORKSPACE: EQUALLINES
  M = NUMBER OF LINES

THIS FUNCTION PERFORMS THE NECESSARY CALCULATIONS TO TEST THE
FOLLOWING THREE HYPOTHESES:
   1) BETA[1]=BETA[2]=∘∘∘=BETA
   2) ALPHA[1]=ALPHA[2]=∘∘∘=ALPHA
   3) HYPOTHESES 1 AND 2 SIMULTANEOUSLY
OUTPUT INCLUDES THE BETA[I'S], THE COMMON BETA, THE ALPHA[I'S],
AND THE COMMON ALPHA.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

OUR DATA S IS:
 S1 EQUALS:
  3.5    4.1    4.4    5      5.5    6.1    6.6
 24     32     37     40     43     51     62
S2 EQUALS:
  3.2    3.9    4.9    6.1    7      8.1
 22     33     39     44     53     57
S3 EQUALS:
  3      4      5      6      6.5    7      7.3    7.4
 32     36     47     49     55     59     64     64
WHERE S1, S2, S3 CORRESPOND TO THE INDEPENDENT AND
DEPENDENT DATA FOR LINE 1, 2, AND 3 RESPECTIVELY

WE WILL NOW EXECUTE THE PROGRAM.  AS INQUIRIES ARE MADE TYPE
IN THE APPROPRIATE RESPONSE...
FOR INSTANCE, WHEN DATA SET 1 IS ASKED FOR SIMPLY TYPE IN THE
APPROPRIATE VARIABLE NAME, S1.

NOW TO EXECUTE THE PROGRAM WE TYPE:

LINES 3
ENTER DATA SET NUMBER 1...
⎕:
      S1
ENTER DATA SET NUMBER 2...
⎕:
      S2
ENTER DATA SET NUMBER 3...
⎕:
      S3

ANOVA TABLE
SUM OF SQUARES COMMON SLOPE = 2598.65 WITH 1 DEGREE OF FREEDOM.
SUM OF SQUARES EQUAL SLOPES HYPOTHESIS = 96.9385
     WITH 2 DEGREES OF FREEDOM.
SUM OF SQUARES EQUAL INTERCEPTS HYPOTHESIS = 441.516
     WITH 2 DEGREES OF FREEDOM.
```

```
SUM OF SQUARES EQUAL SLOPES AND INTERCEPTS HYPOTHESIS = 291.997
     WITH 4 DEGREES OF FREEDOM.
SUM OF SQUARES ERROR = 96.6641 WITH 15 DEGREES OF FREEDOM.
SUM OF SQUARES TOTAL = 45579

BETA[I'S] EQUAL: 10.9685  6.77352  7.33442
COMMON BETA EQUALS: 7.89619
ALPHA[I'S] EQUAL: ¯13.8701  3.85318  8.39375
COMMON ALPHA EQUALS: 1.81413
```

```
      )COPY 1130 ONEWAYCHISQ
      SHOWME

FUNCTION: P CHISQ1 X          WORKSPACE: ONEWAYCHISQ
  P = VECTOR OF CELL PROBABILITIES (IF P IS ANY SCALAR THE
      TEST ASSUMES EQUAL CELL PROBABILITIES)
  X = VECTOR OF CELL FREQUENCIES

THE FUNCTION COMPUTES A CHI-SQUARE GOODNESS-OF-FIT STATISTIC.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR CHISQ1
 SOURCE: STEEL AND TORRIE, PRINCIPLES AND PROCEDURES OF
         STATISTICS, PAGE 365.

THE CELL OBSERVATIONS:
O ← 1178  291  273  156
THE CELL PROBABILITIES:
P ← 0.5625  0.1875  0.1875  0.0625

TO EXECUTE THE PROGRAM WE TYPE:

P CHISQ1 O
CHI SQUARE EQUALS 54.3129 WITH 3 DEGREES OF FREEDOM.

IF YOU WISH TO SEE THE EXPECTED CELL FREQUENCIES TYPE:

E
1067.63  355.875  355.875  118.625
```

```
      )COPY 1130 TWOWAYCHISQ
      SHOWME

FUNCTION: CHISQ2 X           WORKSPACE: TWOWAYCHISQ
  X = (M,N) OBSERVATION MATRIX
       M = NUMBER OF GROUPS IN FIRST CLASSIFICATION
       N = NUMBER OF GROUPS IN CROSS CLASSIFICATION

THIS FUNCTION COMPUTES A CHI-SQUARE STATISTIC TO TEST FOR
ASSOCIATION BETWEEN THE TWO CLASSIFICATIONS.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR CHISQ2
 SOURCE: ANDERSON AND BANCROFT, STATISTICAL THEORY IN
         RESEARCH, PAGE 138.

OBSERVATION MATRIX, X, EQUALS:

    1178     273
     291     156

TO EXECUTE THE PROGRAM WE TYPE:

CHISQ2 X
CHI SQUARE EQUALS 50.5381 WITH 1 DEGREES OF FREEDOM.

IF YOU WISH TO SEE THE EXPECTED CELL FREQUENCIES TYPE:

E

 1123.03      327.966
 345.966      101.034
```

```
      )COPY 1130 THREEWAYCHI
      SHOWME

FUNCTION: K CHISQ3 X          WORKSPACE: THREEWAYCHI
  K = NUMBER OF GROUPS IN THIRD CLASSIFICATION (LAYERS)
  X = (M,N) OBSERVATION MATRIX
       M = K TIMES R
       R = NUMBER OF GROUPS IN SECOND CLASSIFICATION(ROWS)
       N = NUMBER OF GROUPS IN FIRST CLASSIFICATION(COLUMNS)

THIS FUNCTION DOES THE NECESSARY CALCULATIONS FOR TESTING FOR
ASSOCIATION BETWEEN THE TWO WAY CLASSIFICATIONS AND ASSOCIATION
BETWEEN THE THREE WAY CLASSIFICATIONS.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR CHISQ3
 SOURCE: KENDALL AND STUART, THE ADVANCED THEORY OF STATISTICS
         VOL. 2, PAGE 581.

THE OBSERVATION MATRIX, X, FOR THE 2×2×2 CONTINGENCY TABLE
EQUALS:

  427  509
  494  462
  460  440
  475  467

TO EXECUTE THE PROGRAM WE TYPE:

 2 CHISQ3 X

CHI SQUARE TABLE
R×C = 1.61742 WITH 1 DEGREES OF FREEDOM
R×L = 0.139649 WITH 1 DEGREES OF FREEDOM
C×L = 2.80715 WITH 1 DEGREES OF FREEDOM
R×C×L = 4.23963 WITH 1 DEGREES OF FREEDOM
```

```
      )COPY 1130 COVARIANCE
      SHOWME

FUNCTION: ANACOV K             WORKSPACE: COVARIANCE
  K = NUMBER OF TREATMENT LEVELS

FUNCTION: DUNCOV               WORKSPACE: COVARIANCE
```

'ANACOV' DOES THE NECESSARY CALCULATIONS FOR AN ANALYSIS OF COVARIANCE WITH SINGLE COVARIATE AND SINGLE CLASSIFICATION. THE DATA IS ENTERED TWO ROWS AT A TIME WHERE THE FIRST ROW IS ALWAYS THE INDEPENDENT OBSERVATIONS AND THE SECOND ROW IS THE DEPENDENT. THERE ARE K OF THESE DATA SETS WHICH ARE ENTERED. THE SAMPLE SIZES NEED NOT BE EQUAL.
'DUNCOV' IS USED AS AN OPTION BY 'ANACOV' TO COMPUTE A DUNCAN'S MULTIPLE RANGE TEST ON THE ADJUSTED MEANS. THE USER MUST INPUT THE LEAST SIGNIFICANT STUDENTIZED RANGE FROM THE AVAILABLE TABLES.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

EXAMPLE FOR ANACOV AND DUNCOV
SOURCE: FRYER, CONCEPTS AND METHODS OF EXPERIMENTAL STATISTICS, PAGE 402.

OUR DATA, S IS:

```
   70   72   75   76   80   81   85   88   91   92   98  100
   72   76   72   76   82   76   80   78   86   86   86   90
   90   93   95   96   97  102  105  108  110  115  117  120
   70   73   68   76   78   72   76   86   80   82   85   90
  115  110  118  122  125  125  130  133  135  119  140  140
   76   73   72   82   78   87   85   88   82   81   88   92
```

THE MATRIX, S, HAS AS ITS FIRST TWO ROWS THE INDEPENDENT AND DEPENDENT DATA FOR TREATMENT LEVEL 1. ROWS 3, 4 AND 5, 6 ARE SET UP THE SAME WAY FOR LEVELS 2 AND 3.
WHEN THE PROGRAM ASKS FOR DATA SET 1, ALL THAT IS NECESSARY TO ENTER IS THE VARIABLE NAME FOR THOSE TWO ROWS, S[1 2;]. FOR DATA SET 2, ENTER S[3 4;], ETC.
IF THE DATA SETS WERE OF UNEQUAL SIZE A DIFFERENT VARIABLE NAME WOULD HAVE TO BE GIVEN TO EACH TWO ROW DATA SET MATRIX.

WE WILL USE THE OPTION OF DOING THE DUNCAN'S MULTIPLE RANGE TEST AND WILL ENTER 2.89 3.04 3.12 FOR THE LEAST SIGNIFICANT STUDENTIZED RANGE.

NOW TO EXECUTE THE PROGRAM WE TYPE:

```
ANACOV 3
ENTER X-Y DATA SET FOR TREATMENT 1
⎕:
      S[1 2;]
```

```
ENTER X-Y DATA SET FOR TREATMENT 2
⎕:
      S[3 4;]
ENTER X-Y DATA SET FOR TREATMENT 3
⎕:
      S[5 6;]

ANALYSIS OF COVARIANCE
SOURCE
   TREATMENT SUM OF SQUARES -- D.F. = 2
      FOR X  : 10591.3
      FOR XY : 528
      FOR Y  : 96
   ERROR SUM OF SQUARES -- D.F. = 33
      FOR X  : 3252.25
      FOR XY : 1822
      FOR Y  : 1341.94
      DEVIATION FROM REGRESSION 321.202 D.F. = 32
   TOTAL SUM OF SQUARES -- D.F. = 35
      FOR X  : 13843.6
      FOR XY : 2350
      FOR Y  : 1437.94
      DEVIATION FROM REGRESSION 1039.02 D.F. = 34
   ADJUSTED TREATMENT MEAN SQUARE 358.907  D.F. = 2
F(2,32) = 35.7563


DO YOU WISH TO COMPUTE DUNCAN'S MULTIPLE RANGE 'TEST' ON ADJUSTED
MEANS OR 'NO' TEST?
TEST
ENTER APPROPRIATE VALUES FOR LEAST SIGNIFICANT STUDENTIZED RANGE:
⎕:
      2.89 3.04 3.12

ADJUSTED MEANS:
 91.578        78.3735       70.0485
 1             2             3

91.578 IS SIGNIFICANTLY DIFFERENT FROM 78.3735  70.0485
78.3735 IS SIGNIFICANTLY DIFFERENT FROM 70.0485
```

```
      )COPY 1130 UNEQUALSIZE:ANOVA2
      SHOWME

FUNCTION:M UNNOVA N          WORKSPACE: UNEQUALSIZE:ANOVA2
  M = NUMBER OF ROW TREATMENT LEVELS
  N = NUMBER OF COLUMN TREATMENT LEVELS

THE FUNCTION DOES THE NECESSARY CALCULATIONS FOR A TWO-WAY
ANALYSIS OF VARIANCE WHEN THERE ARE UNEQUAL CELL SIZES.
THE DATA FROM EACH CELL WILL BE ASKED FOR ACCORDING TO THE
SUBSCRIPT OF THAT CELL.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR UNNOVA
 SOURCE: BANCROFT, TOPICS IN INTERMEDIATE STATISTICAL
         METHODS, VOL. 1, PAGE 20 AND 24.

OUR DATA IS:        COLUMNS
              A           B
        A     22,25       ¯1,40,18
        B     41,41       23,13
    ROWS C    29,20,37    ---
        D     49,50       61
        E     55          ---

THE DATA WILL BE ENTERED INTO THE PROGRAM BY CELLS.  FOR THE
3;1 CELL, FOR INSTANCE, ENTER 29 20 37. FOR THE EMPTY CELLS
ENTER THE WORD SPACE.

NOW TO EXECUTE THE PROGRAM WE TYPE:

5 UNNOVA 2

ENTER DATA FROM CELL[1;1]...
⎕:
      22 25
ENTER DATA FROM CELL[1;2]...
⎕:
      ¯1 40 18
ENTER DATA FROM CELL[2;1]...
⎕:
      41 41
ENTER DATA FROM CELL[2;2]...
⎕:
      23 13
ENTER DATA FROM CELL[3;1]...
⎕:
      29 20 37
ENTER DATA FROM CELL[3;2]...
⎕:
      SPACE
ENTER DATA FROM CELL[4;1]...
⎕:
```

```
      49 50
ENTER DATA FROM CELL[4;2]...
⎕:
      61
ENTER DATA FROM CELL[5;1]...
⎕:
      55
ENTER DATA FROM CELL[5;2]...
⎕:
      SPACE
ANOVA TABLE
SUM OF SQUARES ROWS (ADJUSTED FOR COLUMNS) = 2249.05 WITH 4
        DEGREES OF FREEDOM.
SUM OF SQUARES COLUMNS (ADJUSTED FOR ROWS) = 149.957 WITH 1
         DEGREES OF FREEDOM.
SUM OF SQUARES INTERACTION = 493.5 WITH 2 DEGREES OF FREEDOM.
SUM OF SQUARES ERROR = 1039.66 WITH 4 DEGREES OF FREEDOM.
SUM OF SQUARES TOTAL = 4255.43 WITH 15 DEGREES OF FREEDOM.
```

```
      )COPY 1130 BALINCOMPLT:BLOCK
      SHOWME

FUNCTION: CODE BIB DATA      WORKSPACE: INCOMPLETE:BLOCK
  CODE = A SCALAR WHICH WILL BE USED IN THE DATA MATRIX TO
         INDICATE A MISSING ELEMENT IN THE INCOMPLETE DESIGN
  DATA = (B×T) DATA MATRIX
         B = NUMBER OF BLOCKS
         T = NUMBER OF TREATMENT LEVELS

THIS FUNCTION DOES THE NECESSARY CALCULATIONS FOR AN ANALYSIS
OF VARIANCE WHEN THE BALANCED INCOMPLETE BLOCKS DESIGN IS
USED TO COMPARE T TREATMENTS.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

 EXAMPLE FOR BIB
 SOURCE: HICKS, FUNDAMENTAL CONCEPTS IN THE DESIGN OF
         EXPERIMENTS, PAGE 57.

OUR DATA IS:           TREATMENTS
      BLOCKS       A    B    C    D
         1         2    -   20    7
         2         -   32   14    3
         3         4   13   31    -
         4         0   23    -   11

SINCE 99 IS NOT A LEGITIMATE OBSERVATION LET'S USE IT
TO INDICATE THE MISSING VALUES.
THE DATA MATRIX, M, NOW EQUALS

    2   99   20    7
   99   32   14    3
    4   13   31   99
    0   23   99   11

IN OUR CASE:
NUMBER OF TREATMENT LEVELS IS 4
NUMBER OF BLOCKS IS 4
BLOCK SIZE IS 3
NUMBER OF TIMES EACH TREATMENT APPEARS IS 3

TO EXECUTE THE PROGRAM WE TYPE:

99 BIB M
ENTER NUMBER OF TREATMENT LEVELS:
⎕:
      4
ENTER NUMBER OF BLOCKS:
⎕:
      4
ENTER BLOCK SIZE:
⎕:
      3
```

```
ENTER NUMBER OF TIMES EACH TREATMENT APPEARS:
⎕:
      3

ANOVA TABLE
SUM OF SQUARES BLOCKS = 100.665 WITH 3 DEGREES OF FREEDOM.
SUM OF SQUARES TREATMENTS (ADJUSTED) = 880.833 WITH 3 DEGREES
                 OF FREEDOM.
SUM OF SQUARES ERROR = 363.166 WITH 5 DEGREES OF FREEDOM.
SUM OF SQUARES TOTAL = 1344.66 WITH 11 DEGREES OF FREEDOM.
```

```
      )COPY 1130 GENERALREGR
      SHOWME
```

FUNCTION: X GRST Y   WORKSPACE: GENERALREGR
  X = MATRIX OF THE LEVELS OF THE INDEPENDENT VARIABLES
  Y = OBSERVATION VECTOR

THIS FUNCTION COMPUTES THE BETA VECTOR, HYPOTHESIS SUM OF SQUARES, AND TOTAL UNCORRECTED (Y'Y) SUM OF SQUARES.

DO YOU WISH TO SEE AN 'EXAMPLE' OR 'NO' EXAMPLE?
EXAMPLE

LET US CONSIDER TWO EXAMPLES. THE FIRST IS FROM THE TWO-WAY ANALYSIS OF VARIANCE 'SHOWME'. THE SECOND IS FROM THE MULTIPLE REGRESSION 'SHOWME'. WE WILL OBTAIN SIMILAR RESULTS USING THE GENERAL REGRESSION SIGNIFICANCE TEST.

AN ORTHOGONAL X MATRIX FOR THE TWO-WAY ANOVA, S, EQUALS:

```
 1           0.5          0.5          0.333333
 1           0.5          0.5          0.333333
 1           0.5          0.5          0.333333
 1           0.5          0           ¯0.666667
 1           0.5          0           ¯0.666667
 1           0.5          0           ¯0.666667
 1           0.5         ¯0.5          0.333333
 1           0.5         ¯0.5          0.333333
 1           0.5         ¯0.5          0.333333
 1          ¯0.5          0.5          0.333333
 1          ¯0.5          0.5          0.333333
 1          ¯0.5          0.5          0.333333
 1          ¯0.5          0           ¯0.666667
 1          ¯0.5          0           ¯0.666667
 1          ¯0.5          0           ¯0.666667
 1          ¯0.5         ¯0.5          0.333333
 1          ¯0.5         ¯0.5          0.333333
 1          ¯0.5         ¯0.5          0.333333
```

THE OBSERVATION VECTOR, Y, EQUALS:

```
4  7  5  2  3  2  5  6  4  9  8  8  8  7  5  10  8  7
```

IT IS HELPFUL IN THE FORMING OF THE REDUCED X MATRIX TO KNOW OF THE REDUCTION SYMBOL,/ . THIS OPERATOR IS USED TO ELIMINATE SPECIFIED COLUMNS OF A MATRIX. TO ELIMINATE THE FIRST COLUMN OF A FOUR COLUMN MATRIX, M, WE TYPE: 0 1 1 1/M. TO ELIMINATE THE LAST TWO COLUMNS WE TYPE: 1 1 0 0/M . IN EACH CASE A NEW MATRIX IS FORMED FROM M DELETING THE SPECIFIED COLUMNS AND KEEPING THE REST.

AS YOU CAN SEE FROM THE DESIGN MATRIX WE ARE ASSUMING NO INTERACTION.
WE EXECUTE 'GRST' IN THE FOLLOWING WAYS:

S GRST Y

```
BETA: 6  ¯3.55556  0.166667  2.25
HYPOTHESIS SUM OF SQUARES: 725.222
TOTAL SUM OF SQUARES: 744

(1 0 0 0/S) GRST Y
BETA: 6
HYPOTHESIS SUM OF SQUARES: 648
TOTAL SUM OF SQUARES: 744

(0 1 0 0/S) GRST Y
BETA: ¯3.55556
HYPOTHESIS SUM OF SQUARES: 56.8889
TOTAL SUM OF SQUARES: 744

(0 0 1 0/S) GRST Y
BETA: 0.166667
HYPOTHESIS SUM OF SQUARES: 0.0833333
TOTAL SUM OF SQUARES: 744

(0 0 0 1/S) GRST Y
BETA: 2.24999
HYPOTHESIS SUM OF SQUARES: 20.2498
TOTAL SUM OF SQUARES: 744
```

THUS THE NECESSARY INFORMATION IS AVAILABLE AND IT IS LEFT TO THE USER TO CALCULATE THE FINAL ANSWERS.
THE X MATRIX FOR THE MULTIPLE REGRESSION, D, HAS BEEN CORRECTED FOR THE MEAN AND EQUALS:

```
 1          ¯0.228666      0.642333      1.01633
 1           0.941333      0.542334      0.206334
 1           0.0613337    ¯0.547666     ¯0.463666
 1           0.491334     ¯0.577666     ¯0.233666
 1           0.241334      0.292334     ¯1.48367
 1           0.261334     ¯0.0476664    ¯1.89367
 1           0.461334      0.782334     ¯0.843666
 1           0.501334     ¯0.417666     ¯1.42367
 1          ¯0.358666     ¯0.417666      0.786334
 1          ¯0.178666     ¯0.167666      1.50633
 1          ¯0.418666      0.0123335     0.826334
 1          ¯0.498666     ¯0.167666     ¯0.0336657
 1          ¯1.05867       0.0423335    ¯0.163666
 1          ¯0.608666      0.0923336     0.936334
 1          ¯0.158666      0.112334      1.20633
 1          ¯0.248666      0.162334      1.94633
 1          ¯0.828666     ¯0.627667     ¯0.143666
 1           0.841334     ¯0.187667      0.656334
 1           1.33133      ¯0.297666      0.506334
 1           0.661334     ¯0.357666     ¯0.203666
 1           0.841334      0.982334      1.51633
 1          ¯0.348666     ¯0.557666     ¯1.27367
 1          ¯0.618666     ¯0.497666     ¯1.14367
 1          ¯0.108666     ¯0.607666     ¯1.57367
 1          ¯0.488666     ¯0.567666     ¯0.673666
```

1	¯0.668666	¯0.607666	¯1.01367
1	0.461334	1.46233	1.84633
1	¯0.148666	0.672333	¯0.373666
1	0.211334	¯0.557666	0.0563335
1	¯0.338666	1.41233	¯0.0736656

THE OBSERVATION VECTOR, T, EQUALS:

0.34	0.11	0.38	0.68	0.18
0	0.08	0.11	1.53	0.77
1.17	1.01	0.89	1.4	1.05
1.15	1.49	0.51	0.18	0.34
0.36	0.89	0.91	0.92	1.35
1.33	0.23	0.26	0.73	0.23

TO TEST THE SAME HYPOTHESES WE DID IN THE 'MULTIREGR' 'SHOWME' WE EXECUTE 'GRST' IN THE FOLLOWING WAYS:

```
D GRST Y
BETA: 0.686  ¯0.531455  ¯0.439636  0.208975
HYPOTHESIS SUM OF SQUARES: 19.6226
TOTAL SUM OF SQUARES: 20.8074

(1 0 0 0/D) GRST Y
BETA: 0.686
HYPOTHESIS SUM OF SQUARES: 14.1179
TOTAL SUM OF SQUARES: 20.8074

(0 1 1 1/D) GRST Y
BETA: ¯0.531455  ¯0.439635  0.208974
HYPOTHESIS SUM OF SQUARES: 5.50471
TOTAL SUM OF SQUARES: 20.8074

(0 1 1 0/D) GRST Y
BETA: ¯0.528549  ¯0.289964
HYPOTHESIS SUM OF SQUARES: 4.29984
TOTAL SUM OF SQUARES: 20.8074

(0 1 0 0/D) GRST Y
BETA: ¯0.591613
HYPOTHESIS SUM OF SQUARES: 3.446
TOTAL SUM OF SQUARES: 20.8074
```

ALL THE INFORMATION IS NOW AVAILABLE AND AGAIN IT IS LEFT TO THE USER TO OBTAIN THE FINAL ANSWERS.

```
      )COPY 1130 DATAINPUT
      SHOWME

FUNCTION:D1 INPUT D2         WORKSPACE:DATAINPUT
   D1 = NUMBER OF ROWS
   D2 = NUMBER OF COLUMNS

THIS FUNCTION AIDS IN INPUTING DATA FOR LATER CALCULATIONS.
THE FUNCTION WILL CONTINUE ASKING FOR DATA UNTIL THE ARRAY
DEFINED BY D1 AND D2 IS SATISFIED.
FOR AN EXAMPLE CONSULT THE USER'S MANUAL UNDER 'DATA INPUT'.
```

BIBLIOGRAPHY

[1] Bancroft, T.A. (1968) Topics in Intermediate Statistical Methods, Volume One, The Iowa State University Press, Ames, Iowa.

[2] Berry, P. (1969) Student Text APL/1130 Primer, International Business Machines Corporation, New York.

[3] Draper, N.R. and Smith, H. (1968) Applied Regression Analysis, John Wiley and Sons, Inc., New York.

[4] Fryer, H.C. (1966) Concepts and Methods of Experimental Statistics, Allyn and Bacon, Inc., Boston, Massachusetts.

[5] Kendall, M.G. and Stuart, A. (1961) The Advanced Theory of Statistics, Volume 2, Hafner Publishing Company, New York.

[6] Graybill, F.A. (1961) An Introduction to Linear Statistical Models, Volume I, McGraw-Hill Book Company, Inc., New York.

[7] Hicks, C.R. (1964) Fundamental Concepts in the Design of Experiments, Holt, Rinehart and Winston, New York.

[8] International Business Machines Corporation. IBM 1130 Operating Procedures, Form A26-5717-0, 1968

[9] Kramer, C.Y. (1956) "Extension of Multiple Range Tests to Group Means with Unequal Numbers of Replications", Biometrics, Volume 12, pp. 307-310.

[10] Li, C.C. (1964) Introduction to Experimental Statistics, McGraw-Hill Book Company, New York.

[11] Sanders, D.T. (Editor) (1970) "The Story of APL", Computing Report in Science and Engineering, Volume VI, No. 2, pp. 14-18.

[12] Steel R.G. and Torrie, J.H. (1960) Principles and Procedures of Statistics, McGraw-Hill Book Company, Inc., New York.

APPENDIX

```
      )COPY 1130 DESCRIPTION
      ∇LOOK[⎕]∇

    ∇ LOOK X;M;V;RANGE;T;MEDIAN;H;SD;LARGE;SMALL;MM
[1]     ⎕←SPACE←⍳0
[2]     M←(+/X)÷⍴X
[3]     V←(+/(X-M)*2)÷(⍴X)-1
[4]     RANGE←(⌈/X)-⌊/X
[5]     ORD←⌽X[DSORT X]
[6]     →(0=T←2|⍴X)/TEST
[7]     MEDIAN←ORD[0.5+H←0.5×⍴ORD]
[8]     →SDD
[9]     TEST:MEDIAN←(ORD[H+1]+ORD[H←0.5×⍴ORD])÷2
[10]    SDD:SD←V*0.5
[11]    LARGE←⌈/X
[12]    SMALL←⌊/X
[13]    'MEAN = ';M
[14]    'MEDIAN = ';MEDIAN
[15]    M←+/X∘.=X←ORD
[16]    →((⌈/M)≤1)/PUT
[17]    'MODE = ';((⍴(MM/X))⍴(((⌈/M)-1)⍴0),1)/((MM←(⌈/M)=M)/X)

[18]    PUT:'VARIANCE = ';V
[19]    'STANDARD DEVIATION = ';SD
[20]    'LARGEST VALUE = ';LARGE
[21]    'SMALLEST VALUE = ';SMALL
[22]    'RANGE = ';RANGE
[23]    SPACE
[24]    SPACE
    ∇
```

```
      )COPY 1130 EQUALVAR:TTEST
      ∇TTEST1[⎕]∇

    ∇ X1 TTEST1 X2;XBAR;SPACE;T;SP;ANS
[1]   NUM←(⌈/XBAR)-⌊/XBAR←2ρ(MEAN X1),MEAN X2
[2]   SP←((((ρX1)-1)×VAR X1)+(((ρX2)-1)×VAR X2))÷((ρX1)+(ρX
2)-2)
[3]   DEN←(SP*0.5)×((1÷ρX1)+(1÷ρX2))*0.5
[4]   ⎕←SPACE←⍳0
[5]   'T = ';NUM÷DEN
[6]   'DF = ';((ρX1)+(ρX2)-2)
[7]   SPACE
[8]   'DO YOU WANT THE CONFIDENCE ''INTERVAL'' FOR MU1-MU2
OR ''NO'' INTERVAL?'
[9]   ANS←⍞
[10]  →((2=ρANS)∨3=ρANS)/OUT
[11]  'ENTER T-VALUE:'
[12]  T←⎕
[13]  NUM-DEN×T;' <MU1-MU2< ';NUM+DEN×T
[14]  OUT:SPACE
[15]  SPACE
    ∇
```

```
      )COPY 1130 UNEQUALVAR:TTEST
      ∇TTEST2[⎕]∇

    ∇ X1 TTEST2 X2;XBAR;M;N;SPACE;ANS;T
[1]   NUM←(⌈/XBAR)-⌊/XBAR←2ρ(MEAN X1),MEAN X2
[2]   DEN←((M←(VAR X1)÷ρX1)+(N←(VAR X2)÷ρX2))*0.5
[3]   ⎕←SPACE←⍳0
[4]   'T = ';NUM÷DEN
[5]   'THE APPROXIMATE DEGREES OF FREEDOM EQUAL ';⌈((M+N)*2
)÷((M*2)÷(ρX1)-1)+((N*2))÷(ρX2)-1
[6]   SPACE
[7]   'DO YOU WANT THE CONFIDENCE ''INTERVAL'' FOR MU1-MU2
OR ''NO'' INTERVAL?'
[8]   ANS←⍞
[9]   →((2=ρANS)∨3=ρANS)/OUT
[10]  'ENTER T-VALUE:'
[11]  T←⎕
[12]  NUM-T×DEN;' <MU1-MU2< ';NUM+T×DEN
[13]  OUT:SPACE
[14]  SPACE
    ∇
```

```
      )COPY 1130 PAIREDTTEST
      ∇PTTEST[⎕]∇

    ∇ X1 PTTEST X2;D;SPACE;ANS;T
[1]   →(0≠(ρX1)-ρX2)/TEST2
[2]   NUM←MEAN D←X1-X2
[3]   DEN←((VAR D)÷ρX1)*0.5
[4]   ⎕←SPACE←ι0
[5]   'T = ';NUM÷DEN
[6]   'DF = ';(ρX1)-1
[7]   SPACE
[8]   'DO YOU WANT THE CONFIDENCE ''INTERVAL'' FOR MU1-MU2
OR ''NO'' INTERVAL?'
[9]   ANS←⍞
[10]  →((2=ρANS)∨3=ρANS)/OUT
[11]  'ENTER T-VALUE:'
[12]  T←⎕
[13]  NUM-T×DEN;' <MU1-MU2< ';NUM+T×DEN
[14]  SPACE
[15]  SPACE
[16]  →0
[17]  TEST2:'ERROR....N[1]≠N[2]'
[18]  OUT:SPACE
[19]  SPACE
    ∇
```

```
      )COPY 1130 BARTLETT
      ∇BARTLT[⎕]∇

    ∇ BARTLT;INDEX;V;INPUT;HOLD
[1]   ⎕←SPACE←⍳0
[2]   'ENTER NUMBER OF CELLS:'
[3]   DATA←INDEX←V←⍳0
[4]   INDEX←0
[5]   K←⎕+P←0×I←1
[6]   AGAIN:→(P=1)/NODAT
[7]   I←I+1
[8]   'ENTER DATA FROM CELL ';I-1;':'
[9]   INPUT←⎕
[10]  HOLD←DATA←DATA,INPUT
[11]  INDEX←INDEX,⍴INPUT
[12]  →PASS
[13]  NODAT:I←I+1
[14]  PASS:V←V,VAR DATA[(+/INDEX[⍳I-1])+⍳INDEX[I]]
[15]  →(I≠K+1)/AGAIN
[16]  LNV←⍟V
[17]  DF←INDEX[1+⍳(⍴INDEX)-1]-1
[18]  LNPL←⍟PL←(+/DF×V)÷TDF←+/DF
[19]  B←((TDF×LNPL)-(+/DF×LNV))÷1+((+/÷DF)-÷TDF)÷3×(K-1)
[20]  SPACE
[21]  'BARTLETT''S STATISTIC = ';B;' WITH ';K-1;' DEGREES O
F FREEDOM.'
[22]  SPACE
[23]  'DO YOU WISH TO TRY SQUARE ''ROOT'', ''LOG BASE E'',
''ARCSIN'' '
[24]  'TRANSFORMATION OR ''NO'' TRANSFORMATION?'
[25]  ANS←⍞
[26]  →((4=⍴ANS)∨5=⍴ANS)/ROOT
[27]  →((6=⍴ANS)∨7=⍴ANS)/ARC
[28]  →((2=⍴ANS)∨3=⍴ANS)/OUT
[29]  DATA←⍟HOLD+1
[30]  →SET
[31]  ROOT:DATA←HOLD*0.5
[32]  →SET
[33]  ARC:→((+/((⍴HOLD)⍴1)≤|HOLD)=0)/35
[34]  'FOR ARCSIN X -- X MUST BE WITHIN RANGE ¯1<X<1...'
[35]  →22
[36]  DATA←ARCSIN HOLD
[37]  →SET
[38]  SET:INDEX[1]←0×P←I←1
[39]  V←⍳0
[40]  →AGAIN
[41]  OUT:SPACE
    ∇
```

```
      )COPY 1130 MULTIRANGE
      ∇DUNCAN[⎕]∇

    ∇ K DUNCAN MEANS;MSE;RP;RRP;NTN;NPN;TEST;SPACE;DIFVEC;R
HO;SIGDIF;I
[1]    ⍳0
[2]    'ENTER M.S. ERROR:'
[3]    MSE←⎕
[4]    'ENTER APPROPRIATE VALUES FOR LEAST SIGNIFICANT STUDE
NTIZED RANGE:'
[5]    RP←⎕
[6]    RRP←RP×MSE*0.5
[7]    NTN←K∘.×K←K[DSORT MEANS]
[8]    NPN←K∘.+K
[9]    TEST←|MEANS∘.-MEANS←MEANS[K←DSORT MEANS]
[10]   TEST←TEST×(2×NTN÷NPN)*0.5
[11]   ⎕←SPACE←DIFVEC←⍳0
[12]   RHO←I←0
[13]   T←(⍴TEST)[1]⍴1
[14]   UP:→(((M←(⍴TEST)[1])+1)=I←I+1)/ON
[15]   T[I]←0
[16]   DIFVEC←DIFVEC,SIGDIF←(TEST[I;I+⍳M-I]>RRP[⍳M-I])/(T/ME
ANS)
[17]   →(0=⍴SIGDIF)/ON
[18]   RHO←RHO,⍴SIGDIF
[19]   →UP
[20]   ON:I←0
[21]   'MEANS:';(2,M)⍴MEANS,K
[22]   SPACE
[23]   →((+/RHO)=0)/NODIF
[24]   OUT:→((⍴RHO)=I←I+1)/0
[25]   MEANS[I];' IS SIGNIFICANTLY DIFFERENT FROM ';DIFVEC[(
+/RHO[⍳I])+⍳RHO[I+1]]
[26]   →OUT
[27]   NODIF:'NO SIGNIFICANTLY DIFFERENT MEANS...'
[28]   SPACE
[29]   SPACE
    ∇
```

```
      )COPY 1130 ONEWAYANOVA
      ∇ANOVA1[⎕]∇

    ∇ K ANOVA1 X;SSTOT;SSTR;RHO;I;INPUT;N;ERROR;NUM;CC;SCC;TT
;AA
[1]    →((⍴⍴X)>1)/DATAIN
[2]    SSTOT←SSTR←RHO←TT←⍳0×I←0
[3]    PUT:'INPUT DATA FOR TREATMENT ';I←I+1;'...'
[4]    INPUT←⎕
[5]    SSTOT←SSTOT,+/INPUT*2
[6]    SSTR←SSTR,((+/INPUT)*2)
[7]    RHO←RHO,⍴INPUT
[8]    TT←TT,+/INPUT
[9]    →(I<K)/PUT
[10]   N←+/RHO
[11]   AA←(+/TT)*2
[12]   SSTOT←(+/SSTOT)-AA÷N
[13]   SSTR←(+/SSTR÷RHO)-AA÷N
[14]   ERROR←SSTOT-SSTR
[15]   →ANOVA
[16]   DATAIN:SSTOT←(+/(,X)*2)-A←((+/(,X))*2)÷N←K×(⍴X)[1]
[17]   SSTR←((+/(TT←+⌿X)*2)÷(⍴X)[1])-A
[18]   ERROR←SSTOT-SSTR
[19]   RHO←K⍴(⍴X)[1]
[20]   ANOVA:'ANOVA TABLE'
[21]   'SUM OF SQUARES TOTAL = ';SSTOT;' WITH ';N-1;' DEGREES
OF FREEDOM.'
[22]   'SUM OF SQUARES TREATMENTS = ';SSTR;' WITH ';K-1;' DEGR
EES OF FREEDOM.'
[23]   'SUM OF SQUARES ERROR = ';ERROR;' WITH ';N-K;' DEGREES
OF FREEDOM.'
[24]   'DO YOU WISH TO RUN SINGLE DEGREE OF FREEDOM ''CONTRAST
S'' OR ''NO'' CONTRASTS?'
[25]   ANS←⍞
[26]   →((2=⍴ANS)∨3=⍴ANS)/0
[27]   'ENTER NUMBER OF CONTRASTS:'
[28]   I←0×NUM←⎕
[29]   BACK:I←I+1
[30]   'ENTER CONTRAST COEFFICIENT SET NUMBER ';I;'...'
[31]   CC←⎕
[32]   SCC←((+/CC×TT)*2)÷+/RHO×CC*2
[33]   'SUM OF SQUARES FOR CONTRAST ';I;' = ';SCC
[34]   →(I≠NUM)/BACK
    ∇
```

```
      )COPY 1130 TWOWAYANOVA
      ∇ANOVA2[⎕]∇

    ∇ ANOVA2 X;SPACE;N;J;I;SUMS;DATA;SSA;G;SSB;SSC;SSAB;SSTOT
;ANS;C1;C2;CC;TT;KK
[1]   ⎕←SPACE←⍳0
[2]   'ENTER NUMBER OF REPLICATIONS IN EACH CELL'
[3]   N←⎕
[4]   J←((⍴X)[1])÷N
[5]   I←(⍴X)[2]
[6]   SUMS←⍉(I,J)⍴+/(2⍴(I×J),N)⍴⍉X
[7]   DATA←,⍉X
[8]   SSA←((+/(+⌿SUMS)*2)÷(J×N))-G←((+/,SUMS)*2)÷N×I×J
[9]   SSB←((+/(+/SUMS)*2)÷(I×N))-G
[10]  SSC←(+/(,SUMS*2)÷N)-G
[11]  SSAB←SSC-SSA+SSB
[12]  SSTOT←(+/(DATA*2))-G
[13]  SPACE
[14]  'ANOVA TABLE'
[15]  'SUM OF SQUARES TOTAL = ';SSTOT;' WITH ';(I×J×N)-1;' DE
GREES OF FREEDOM'
[16]  'SUM OF SQUARES TREATMENT A = ';SSA;' WITH ';I-1;' DEGR
EES OF FREEDOM'
[17]  'SUM OF SQUARES TREATMENT B = ';SSB;' WITH ';J-1;' DEGR
EES OF FREEDOM'
[18]  'SUM OF SQUARES INTERACTION = ';SSAB;' WITH ';(I-1)×(J-
1);' DEGREES OF FREEDOM'
[19]  'SUM OF SQUARES ERROR = ';SSTOT-SSC;' WITH ';(I×J)×(N-1
);' DEGREES OF FREEDOM'
[20]  CONT:SPACE
[21]  'DO YOU WISH TO DO CONTRASTS ON ''COLUMNS'',''ROWS'', '
'INTERACTION'' '
[22]  'OR ''NO'' CONTRASTS?'
[23]  ANS←⍞
[24]  →((7=⍴ANS)∨8=⍴ANS)/COLUMN
[25]  →((2=⍴ANS)∨3=⍴ANS)/0
[26]  →((4=⍴ANS)∨5=⍴ANS)/ROW
[27]  'ENTER CONTRAST COEFFICIENTS FOR TREATMENT A'
[28]  C1←⎕
[29]  'ENTER CONTRAST COEFFICIENTS FOR TREATMENT B'
[30]  C2←⎕
[31]  CC←,⍉C1∘.×C2
[32]  TT←(,SUMS)+0×KK←N
[33]  →OUTPUT
[34]  COLUMN:'ENTER CONTRAST COEFFICIENTS FOR TREATMENT A'
[35]  CC←⎕
[36]  TT←+⌿SUMS
[37]  KK←J×N
[38]  →OUTPUT
[39]  ROW:'ENTER CONTRAST COEFFICIENTS FOR TREATMENT B'
```

```
[40]  CC←⎕
[41]  TT←+/SUMS
[42]  KK←I×N
[43]  OUTPUT:'SUM OF SQUARES CONTRAST = ';((+/CC×TT)*2)÷(KK×+
/CC*2)
[44]  →CONT
    ∇
```

```
      )COPY 1130 REGRESSION
      ∇REGR[⎕]∇

    ∇ X REGR Y;SXX;SYY;SXY;INT;BETA;ANS;T;BET;MSE;PY;RES;AC
HE;OTPT
[1]    SXX←+/(X-MEAN X)*2
[2]    SYY←+/(Y-MEAN Y)*2
[3]    SXY←+/(X-MEAN X)×(Y-MEAN Y)
[4]    INT←MEAN Y
[5]    BETA←SXY÷SXX
[6]    ⍳0
[7]    'REGRESSION EQUATION:  Y=';INT;'+';BETA;'(X-';MEAN X;
')'
[8]    'CORRELATION COEFFICIENT:';BETA×(SXX÷SYY)*0.5
[9]    ⍳0
[10]   'DO YOU WISH TO TEST BETA EQUAL TO ''BETA NOUGHT'' OR
 ''ZERO'' OR ''NO'' TEST?'
[11]   ANS←⍞
[12]   →((11=ρANS)∨12=ρANS)/BETA0
[13]   →((2=ρANS)∨2=ρANS)/0
[14]   T←BETA÷((MSE←(((SYY-BETA×SXY)÷(ρY)-2)))÷SXX)*0.5
[15]   →OUTPUT
[16]   BETA0:'ENTER VALUE OF BETA NOUGHT:'
[17]   BET←⎕
[18]   T←(BETA-BET)÷((MSE←(((SYY-BETA×SXY)÷(ρY)-2)))÷SXX)*0.
5
[19]   OUTPUT:'T-VALUE = ';T;' WITH ';(ρX)-2;' DEGREES OF FR
EEDOM .'
[20]   →(0≠=/X)/22
[21]   X ERRFIT Y
[22]   ⍳0
[23]   'DO YOU WISH CONFIDENCE ''LIMITS'' OR ''NO'' LIMITS?'

[24]   ANS←⍞
[25]   →((2=ρANS)∨3=ρANS)/NEXT
[26]   'INPUT T-VALUE AND VALUE FOR X:'
[27]   T←⎕
[28]   ⍳0
[29]   'CONFIDENCE INTERVALS'
[30]   '1- ON BETA'
[31]   '       U.L. = ';BETA+T[1]×(MSE÷SXX)*0.5
[32]   '       L.L. = ';BETA-T[1]×(MSE÷SXX)*0.5
[33]   →(T[2]=0)/NEXT
[34]   '2- ON A POINT ON REGRESSION LINE (X= ';T[2];')'
[35]   PY←INT+BETA×(T[2]-MEAN X)
[36]   '       U.L. = ';PY+T[1]×ACHE←((MSE÷(ρX))+(MSE÷SXX)×(T[
2]-MEAN X)*2)*0.5
[37]   '       L.L. = ';PY-T[1]×ACHE
[38]   '3- PREDICTION LIMITS (X= ';T[2];')'
[39]   '       U.L. = ';PY+T[1]×ACHE←((MSE÷(ρX))+(MSE÷SXX)×((T
[2]-MEAN X)*2)+MSE)*0.5
[40]   '       L.L. = ';PY-T[1]×ACHE
[41]   ⍳0
[42]   NEXT:'DO YOU WISH TO HAVE PREDICTIONS AND ''RESIDUALS
'' PRINTED OR ''NOT''?'
```

```
[43]  ANS←⍞
[44]  →((3=⍴ANS)∨4=⍴ANS)/0
[45]  RES←Y-PY←INT+BETA×(X-MEAN X)
[46]  OTPT←(4,(⍴X))⍴X,Y,PY,RES
[47]  ⍳0
[48]  'X, Y, Y-PREDICTED, AND RESIDUALS:'
[49]  ⍉OTPT
    ∇
```

```
      )COPY 1130 MULTIREGR
      ∇MREG[⎕]∇

    ∇ X MREG Y;K;N;G;A;I;J;B;C;SSR;DD;L
[1]   K←(ρX)[2]+0×N←(ρX)[1]
[2]   BETA←(JINV((⍉DD)+.×DD))+.×(⍉DD←X-(N,K)ρ(+⌿X)÷N)+.×Y
[3]   G←(⍉DD)+.×Y
[4]   A←((K),(K+2))ρ0+I←J←0
[5]   UP1:I←I+1
[6]   J←0×L←2
[7]   →(I>K)/ON
[8]   UP2:J←J+1
[9]   →(I>J)/UP2
[10]  →(J>K)/UP1
[11]  A[J;I]←A[I;J]←+/(X[;I]-MEAN X[;I])×(X[;J]-MEAN X[;J])

[12]  →UP2
[13]  ON:A[;K+1]←G
[14]  A[;K+2]←+/A
[15]  '(FORWARD DOOLITTLE METHOD)';A
[16]  B←(ρA)ρ0×I←1
[17]  C←((2×K),(K+2))ρ0
[18]  C[1;]←A[1;]←A[1;]
[19]  C[2;]←B[1;]←A[1;]÷A[1;1]
[20]  OUT:I←I+1
[21]  →(I>K)/FINI
[22]  J←0
[23]  IN:J←J+1
[24]  →(J>K)/OUT2
[25]  A[I;]←A[I;]-A[J;I]×B[J;]
[26]  →IN
[27]  OUT2:C[L←L+1;]←A[I;]
[28]  C[L←L+1;]←B[I;]←A[I;]÷A[I;I]
[29]  →OUT
[30]  FINI:⍳0
[31]  C
[32]  ⍳0
[33]  'ANOVA TABLE'
[34]  'SUM OF SQUARES REGRESSION = ';SSR←+/BETA×G;' WITH ';
K;' DEGREES OF FREEDOM.'
[35]  I←1
[36]  '    SUM OF SQUARES X[';I;'] = ';A[I;K+1]×B[I;K+1];' W
ITH 1 DEGREE OF FREEDOM.'
[37]  OUTCT:I←I+1
[38]  '    SUM OF SQUARES X[';I;']|X[';⍳I-1;'] = ';A[I;K+1]×
B[I;K+1];' WITH 1 DEGREE OF FREEDOM.'
[39]  →(I<K)/OUTCT
[40]  'SUM OF SQUARES RESIDUAL = ';(+/(Y-MEAN Y)*2)-SSR;' W
ITH ';N-K+1;' DEGREES OF FREEDOM.'
[41]  'SUM OF SQUARES TOTAL = ';+/(Y-MEAN Y)*2;' WITH ';N-1
;' DEGREES OF FREEDOM.'
[42]  ⍳0
    ∇
```

```
      )COPY 1130 POLYNOMIAL
      ∇POLYR[⎕]∇

    ∇ N POLYR DATA;Y;X;LIST;X1;D;X2;X3;X4;GAMMA;BETA;SUM;YB
AR;SS;SPACE;K
[1]    X←(DATA[;1])[DSORT DATA[;1]]+0×Y←(DATA[;2])[DSORT DAT
A[;1]]
[2]    LIST←X1←(X←X-MEAN X)÷D←X[1]-X[2]
[3]    →(1=N)/OUT
[4]    LIST←LIST,X2←(X1*2)-((((ρX)*2)-1)÷12)
[5]    →(2=N)/OUT
[6]    LIST←LIST,X3←(X1*3)-X1×((3×(ρX)*2)-7)÷20
[7]    →(0=N)/OUT
[8]    LIST←LIST,X4←((X1*4)-((((3×(ρX)*2)-13)÷14)×X1*2))+(3÷
560)×(((ρX)*2)-1)×((ρX)*2)-9
[9]    OUT:LIST←⍉(N,(ρX))ρLIST
[10]   GAMMA←(⍉LIST)+.×Y←((ρX),1)ρY←(Y-YBAR←MEAN Y)
[11]   BETA←GAMMA÷SUM←(ρGAMMA)ρ+/LIST*2
[12]   SS←,BETA×GAMMA
[13]   ⎕←SPACE←⍳0
[14]   SPACE
[15]   'THE BETA VECTOR EQUALS: '
[16]   (1+ρ,BETA)ρ(YBAR),,BETA
[17]   K←0
[18]   SPACE
[19]   'ANOVA TABLE'
[20]   BACK:K←K+1
[21]   'SUM OF SQUARES BETA[';K;'] = ';SS[K];' WITH 1 DEGREE
 OF FREEDOM.'
[22]   →(K<N)/BACK
[23]   'SUM OF SQUARES RESIDUAL = ';(+/((,Y)-MEAN,Y)*2)-+/SS
;' WITH ';(ρ,Y)-K+1;' DEGREES OF FREEDOM.'
[24]   'SUM OF SQUARES TOTAL = ';(+/((,Y)-MEAN,Y)*2);' WITH
';(ρ,Y)-1;' DEGREES OF FREEDOM.'
    ∇
```

```
      )COPY 1130 EQUALLINES
      ∇LINES[⎕]∇

    ∇ LINES M;MEANX;MEANY;SXX;SYY;SXY;SUMY;RHO;YYY;I;THIS;T
OTAL;BETA;THAT;THOSE;XX;YY;XXX;SET;ANS;SSXX;SSXY;BB
    ETA
[1]    MEANX←MEANY←SXX←SYY←SXY←SUMY←RHO←YYY←XXX←⍳0×I←0
[2]    INPUT:'ENTER DATA SET NUMBER ';I←I+1;'...'
[3]    SET←⎕
[4]    XX←SET[1;]
[5]    YY←SET[2;]
[6]    SXX←SXX,+/(XX-MEAN XX)*2
[7]    SYY←SYY,+/(YY-MEAN YY)*2
[8]    SXY←SXY,+/(XX-MEAN XX)×(YY-MEAN YY)
[9]    SUMY←SUMY,+/YY
[10]   MEANY←MEANY,MEAN YY
[11]   MEANX←MEANX,MEAN XX
[12]   XXX←XXX,XX
[13]   YYY←YYY,YY
[14]   RHO←RHO,⍴YY
[15]   →(I≠M)/INPUT
[16]   BETA←SXY÷SXX
[17]   TOTAL←YYY+.×YYY
[18]   SSXX←+/(XXX-MEAN XXX)*2
[19]   SSXY←+/(XXX-MEAN XXX)×(YYY-MEAN YYY)
[20]   BBETA←SSXY÷SSXX
[21]   ⍳0
[22]   'ANOVA TABLE'
[23]   'SUM OF SQUARES COMMON SLOPE = ';THIS←((+/SXY)*2)÷+/S
XX;' WITH ';1;' DEGREE OF FREEDOM.'
[24]   'SUM OF SQUARES EQUAL SLOPES HYPOTHESIS = ';THAT←(+/B
ETA×SXY)-THIS
[25]   '       WITH ';M-1;' DEGREES OF FREEDOM.'
[26]   'SUM OF SQUARES EQUAL INTERCEPTS HYPOTHESIS = ';THOSE
←(+/(SUMY*2)÷RHO)-((+/SUMY)*2)÷+/RHO
[27]   '       WITH ';M-1;' DEGREES OF FREEDOM.'
[28]   'SUM OF SQUARES EQUAL SLOPES AND INTERCEPTS HYPOTHESI
S = ';(+/BETA×SXY)+THOSE-BBETA×SSXY
[29]   '       WITH ';2×M-1;' DEGREES OF FREEDOM.'
[30]   'SUM OF SQUARES ERROR = ';TOTAL-(+/(SUMY*2)÷RHO)++/BE
TA×SXY;' WITH ';(+/RHO)-2×M;' DEGREES OF FREEDOM.'
[31]   'SUM OF SQUARES TOTAL = ';TOTAL
[32]   ⍳0
[33]   'BETA[I''S] EQUAL: ';BETA
[34]   'COMMON BETA EQUALS: ';BBETA
[35]   'ALPHA[I''S] EQUAL: ';MEANY-BETA×MEANX
[36]   'COMMON ALPHA EQUALS: ';(MEAN YYY)-BBETA×MEAN XXX
    ∇
```

```
      )COPY 1130 ONEWAYCHISQ
      ∇CHISQ1[⎕]∇

    ∇ P CHISQ1 X
[1]   →((ρP)>1)/ON
[2]   E←(ρX)ρMEAN X
[3]   →OVER
[4]   ON:E←P×+/X
[5]   OVER:CHI←+/((X-E)*2)÷E
[6]   'CHI SQUARE EQUALS ';CHI;' WITH ';×/(ρX)-1;' DEGREES
OF FREEDOM.'
    ∇
```

```
      )COPY 1130 TWOWAYCHISQ
      ∇CHISQ2[⎕]∇

    ∇ CHISQ2 X
[1]   E←((+/X)∘.×(+⌿X))÷+/,X
[2]   CHI←+/,((X-E)*2)÷E
[3]   'CHI SQUARE EQUALS ';CHI;' WITH ';×/(⍴X)-1;' DEGREES
OF FREEDOM.'
    ∇
```

```
      )COPY 1130 THREEWAYCHI
      ∇∇CHISQ3[⎕]∇

    ∇ K CHISQ3 X;N;M;CELLS;ROWS;COLS;LAYERS;E3;CHI3;E21;CHI
21;CHI22;CHI23;E22;E23;ACROW
[1]   N←(M←(ρX)[1])÷K
[2]   CELLS←((N,M)ρ(Kρ1),(Mρ0))+.×X
[3]   ROWS←+/CELLS
[4]   COLS←+⌿CELLS
[5]   LAYERS←+⌿XR←(K,(ρX)[2])ρ+/X
[6]   E3←((,ROWS∘.×LAYERS)∘.×COLS)÷(+/,X)*2
[7]   CHI3←+/,((X-E3)*2)÷E3
[8]   E21←(ROWS∘.×COLS)÷T←+/,X
[9]   CHI21←+/,((CELLS-E21)*2)÷E21
[10]  E22←(ROWS∘.×LAYERS)÷T
[11]  CHI22←+/,((XR-E22)*2)÷E22
[12]  E23←⍉(COLS∘.×LAYERS)÷T
[13]  ACROW←X[⍳K;]+X[K+⍳K;]
[14]  CHI23←+/,((ACROW-E23)*2)÷E23
[15]  ⍳0
[16]  'CHI SQUARE TABLE'
[17]  'R×C = ';CHI21;' WITH ';((ρX)[2]-1)×(N-1);' DEGREES O
F FREEDOM'
[18]  'R×L = ';CHI22;' WITH ';(K-1)×N-1;' DEGREES OF FREEDO
M'
[19]  'C×L = ';CHI23;' WITH ';((ρX)[2]-1)×(K-1);' DEGREES O
F FREEDOM'
[20]  'R×C×L = ';CHI3-CHI21+CHI22+CHI23;' WITH ';((ρX)[2]-1
)×(K-1)×N-1;' DEGREES OF FREEDOM'
[21]  ⍳0
    ∇
```

```
      )COPY 1130 COVARIANCE
      ∇ANACOV[⎕]∇

    ∇ ANACOV K;X;Y;RHO;TXX;SXX;TX;XMEANS;YMEANS;TYY;SYY;TY;
TXY;SXY;I;DATA;THIS;EXX;EYY;EXY;SSE;RHOTOT
[1]   RHO←TXX←SXX←TX←XMEANS←YMEANS←TYY←SYY←TY←TXY←SXY←⍳0×I←
0
[2]   MORE:'ENTER X-Y DATA SET FOR TREATMENT ';I←I+1
[3]   X←DATA[1;]+0×Y←(DATA←⎕)[2;]
[4]   RHO←RHO,THIS←(⍴DATA)[2]
[5]   TXX←TXX,((+/X)*2)÷THIS
[6]   SXX←SXX,+/X*2
[7]   TX←TX,+/X
[8]   XMEANS←XMEANS,(+/X)÷THIS
[9]   YMEANS←YMEANS,(+/Y)÷THIS
[10]  TYY←TYY,((+/Y)*2)÷THIS
[11]  SYY←SYY,+/Y*2
[12]  TY←TY,+/Y
[13]  TXY←TXY,(+/X)×(+/Y)÷THIS
[14]  SXY←SXY,+/X×Y
[15]  →(I≠K)/MORE
[16]  TX←+/TX+0×TY←+/TY
[17]  TXX←(+/TXX)-(TX*2)÷RHOTOT←+/RHO
[18]  SXX←(+/SXX)-(TX*2)÷RHOTOT
[19]  EXX←(+/SXX)-(+/TXX)
[20]  TYY←(+/TYY)-(TY*2)÷RHOTOT
[21]  SYY←(+/SYY)-(TY*2)÷RHOTOT
[22]  EYY←(+/SYY)-(+/TYY)
[23]  TXY←(+/TXY)-(TX×TY)÷RHOTOT
[24]  SXY←(+/SXY)-(TX×TY)÷RHOTOT
[25]  EXY←(+/SXY)-(+/TXY)
[26]  ⎕←SPACE←⍳0
[27]  'ANALYSIS OF COVARIANCE'
[28]  'SOURCE'
[29]  '    TREATMENT SUM OF SQUARES -- D.F. = ';K-1
[30]  '       FOR X  : ';TXX
[31]  '       FOR XY : ';TXY
[32]  '       FOR Y  : ';TYY
[33]  '    ERROR SUM OF SQUARES -- D.F. = ';RHOTOT-K
[34]  '       FOR X  : ';EXX
[35]  '       FOR XY : ';EXY
[36]  '       FOR Y  : ';EYY
[37]  '       DEVIATION FROM REGRESSION ';SSE←EYY-(EXY*2)÷EX
X;' D.F. = ';RHOTOT-K+1
[38]  '    TOTAL SUM OF SQUARES -- D.F. = ';RHOTOT-1
[39]  '       FOR X  : ';SXX
[40]  '       FOR XY : ';SXY
[41]  '       FOR Y  : ';SYY
[42]  '       DEVIATION FROM REGRESSION ';SSTE←SYY-(SXY*2)÷S
XX;' D.F. = ';RHOTOT-2
[43]  '    ADJUSTED TREATMENT MEAN SQUARE ';(SSTE-SSE)÷K-1;'
  D.F. = ';K-1
[44]  'F(';K-1;',';RHOTOT-K+1;') = ';((SSTE-SSE)÷K-1)÷(SSE÷
RHOTOT-K+1)
```

```
[45]  SPACE
[46]  SPACE
[47]  'DO YOU WISH TO COMPUTE DUNCAN''S MULTIPLE RANGE ''TE
ST'' ON ADJUSTED '
[48]  'MEANS OR ''NO'' TEST?'
[49]  →((2=ρANS)∧3=ρANS←⍞)/0
[50]  DUNCOV
    ∇
```

```
      )COPY 1130 COVARIANCE
      ∇DUNCOV[⎕]∇

    ∇ DUNCOV;RP;RRP;MSE;NTN;NPN;MEANS;L;TEST;XMX;NN;DIFVEC;
SPACE;T;M;SIGDIF
[1]    SPACE
[2]    'ENTER APPROPRIATE VALUES FOR LEAST SIGNIFICANT STUDE
NTIZED RANGE:'
[3]    RP←⎕
[4]    RRP←RP×(MSE←SSE÷RHOTOT-K+1)*0.5
[5]    NTN←RHO∘.×RHO
[6]    NPN←RHO∘.+RHO
[7]    MEANS←YMEANS-(EXY÷EXX)×(XMEANS-TX÷RHOTOT)
[8]    TEST←|MEANS∘.-MEANS←MEANS[L←DSORT MEANS]
[9]    XMX←(XMEANS∘.-XMEANS←XMEANS[L])*2
[10]   NN←NPN÷NTN
[11]   TEST←TEST×(2÷NN+XMX÷EXX)*0.5
[12]   ⎕←DIFVEC←SPACE
[13]   RHO←I←0
[14]   T←(⍴TEST)[1]⍴1
[15]   UP:→(((M←(⍴TEST)[1])+1)=I←I+1)/ON
[16]   T[I]←0
[17]   DIFVEC←DIFVEC,SIGDIF←(TEST[I;I+⍳M-I]>RRP[⍳M-I])/(T/ME
ANS)
[18]   →(0=⍴SIGDIF)/ON
[19]   RHO←RHO,⍴SIGDIF
[20]   →UP
[21]   ON:I←0
[22]   'ADJUSTED MEANS:';(2,M)⍴MEANS,L
[23]   SPACE
[24]   →((+/RHO)=0)/NODIF
[25]   OUT:→((⍴RHO)=I←I+1)/0
[26]   MEANS[I];' IS SIGNIFICANTLY DIFFERENT FROM ';DIFVEC[(
+/RHO[⍳I])+⍳RHO[I+1]]
[27]   →OUT
[28]   NODIF:'NO SIGNIFICANTLY DIFFERENT MEANS...'
[29]   SPACE
[30]   SPACE
    ∇
```

```
      )COPY 1130 UNEQUALSIZE:ANOVA2
      ∇UNNOVA[⎕]∇

    ∇ M UNNOVA N;CELL;CELL2;NN;I;J;INPUT;DATA;N1;N2;A;B;D1;
D2;INVD1;INVD2;C;Q;TAU;SSRA;SSC;SUBTOT;SSINTE;ERR
[1]    ⎕←SPACE←⍳0
[2]    CELL←CELL2←NN←(M,N)⍴0
[3]    I←0
[4]    ADDI:I←I+1
[5]    J←0
[6]    ADDJ:J←J+1
[7]    'ENTER DATA FROM CELL[';I;';';J;']...'
[8]    INPUT←⎕
[9]    CELL[I;J]←+/INPUT
[10]   CELL2[I;J]←+/INPUT*2
[11]   →(0≠⍴⍴INPUT)/ON
[12]   NN[I;J]←1
[13]   →OVER
[14]   ON:NN[I;J]←⍴INPUT
[15]   OVER:→(J≠N)/ADDJ
[16]   →(I≠M)/ADDI
[17]   DATA←CELL
[18]   N1←+/NN
[19]   N2←+⌿NN
[20]   A←+/DATA
[21]   B←+⌿DATA
[22]   INVD1←D1←(2⍴J)⍴1,(J←⍴N1)⍴0
[23]   INVD2←D2←(2⍴J)⍴1,(J←⍴N2)⍴0
[24]   I←0
[25]   BACK1:INVD1[I;]←D1[I←I+1;]×÷N1
[26]   D1[I;]←D1[I;]×N1
[27]   →(I≠⍴N1)/BACK1
[28]   I←0
[29]   BACK2:INVD2[I;]←D2[I←I+1;]×÷N2
[30]   →(I≠⍴N2)/BACK2
[31]   C←D1-NN+.×(INVD2)+.×⍉NN
[32]   Q←A-NN+.×(INVD2)+.×B
[33]   C←(L←((((⍴C)[1]-1)⍴1),0))⌿C
[34]   C←L/C
[35]   Q←L/Q
[36]   TAU←(JINV C)+.×Q
[37]   SSRA←(⍉TAU)+.×Q
[38]   SSC←(+/(B*2)÷N2)-((+/,DATA)*2)÷+/N1
[39]   SUBTOT←(+/,(DATA*2)÷NN)-((+/,DATA)*2)÷+/N1
[40]   SSINTE←SUBTOT-SSC+SSRA
[41]   ERR←((+/,CELL2)-((+/,DATA)*2)÷+/N1)-SUBTOT
[42]   TOT←(+/,CELL2)-((+/,DATA)*2)÷+/N1
[43]   SSCA←SSC+SSRA-(+/(A*2)÷N1)-((+/,DATA)*2)÷+/N2
[44]   'ANOVA TABLE'
[45]   'SUM OF SQUARES ROWS (ADJUSTED FOR COLUMNS) = ';SSRA;
' WITH ';M-1;' DEGREES OF FREEDOM.'
[46]   'SUM OF SQUARES COLUMNS (ADJUSTED FOR ROWS) = ';SSCA;
' WITH ';N-1;' DEGREES OF FREEDOM.'
```

```
[47]  JJ←+/,(0=NN)
[48]  'SUM OF SQUARES INTERACTION = ';SSINTE;' WITH ';((M-1
)×(N-1))-JJ;' DEGREES OF FREEDOM.'
[49]  'SUM OF SQUARES ERROR = ';ERR;' WITH ';(+/,NN-1)-JJ;'
 DEGREES OF FREEDOM.'
[50]  'SUM OF SQUARES TOTAL = ';TOT;' WITH ';(+/,NN)-1;' DE
GREES OF FREEDOM.'
    ∇
```

```
      )COPY 1130 BALINCOMPLT:BLOCK
      ∇BIB[⎕]∇

    ∇ CODE BIB DATA;T;B;K;R;INDEX;DATA1;TOT;SSTOT;SSBLK;TB;I;
P;COLS;LAMBDA;SSTR;SPACE;ANS
[1]    'ENTER NUMBER OF TREATMENT LEVELS:'
[2]    T←⎕
[3]    'ENTER NUMBER OF BLOCKS:'
[4]    B←⎕
[5]    'ENTER BLOCK SIZE:'
[6]    K←⎕
[7]    'ENTER NUMBER OF TIMES EACH TREATMENT APPEARS:'
[8]    R←⎕
[9]    INDEX←~CODE=DATA
[10]   DATA1←INDEX×DATA
[11]   SSTOT←(+/,DATA1*2)-TOT←((+/,DATA1)*2)÷B×K
[12]   SSBLK←(+/((+/DATA1)*2)÷K)-TOT
[13]   TB←⍳0
[14]   I←0
[15]   BACK:P←(T,T)⍴1
[16]   P[;I←I+1]←T⍴0
[17]   COLS←+⌿DATA1
[18]   TB←TB,COLS[I]-(÷K)×+/((~×/P=INDEX)/+/DATA1)
[19]   →(I≠T)/BACK
[20]   LAMBDA←(R×(K-1))÷T-1
[21]   SSTR←(K÷LAMBDA×T)×+/TB*2
[22]   ⎕←SPACE←⍳0
[23]   'ANOVA TABLE'
[24]   'SUM OF SQUARES BLOCKS = ';SSBLK;' WITH ';B-1;' DEGREES
 OF FREEDOM.'
[25]   'SUM OF SQUARES TREATMENTS (ADJUSTED) = ';SSTR;' WITH '
;T-1;' DEGREES OF FREEDOM.'
[26]   'SUM OF SQUARES ERROR = ';SSTOT-SSBLK+SSTR;' WITH ';(+/
,INDEX)+1-B+T;' DEGREES OF FREEDOM.'
[27]   'SUM OF SQUARES TOTAL = ';SSTOT;' WITH ';(+/,INDEX)-1;'
 DEGREES OF FREEDOM.'
    ∇
```

```
      )COPY 1130 GENERALREGR
      ∇GRST[⎕]∇

    ∇ X GRST Y
[1]   'BETA: ';BETA←(JINV(⍉X)+.×X)+.×(⍉X)+.×Y
[2]   'HYPOTHESIS SUM OF SQUARES: ';BETA+.×(⍉X)+.×Y
[3]   'TOTAL SUM OF SQUARES: ';Y+.×Y
[4]   ⍳0
    ∇
```

```
      )COPY 1130 DATAINPUT
      ∇INPUT[⎕]∇

    ∇ Z←D1 INPUT D2
[1]   Z←⍳0
[2]   BACK:Z←Z,⎕
[3]   →((⍴Z)<D1×D2)/BACK
[4]   →(D1=1)/VECTOR
[5]   Z←(D1,D2)⍴Z
[6]   →0
[7]   VECTOR:Z←(D2)⍴Z
    ∇
```

```
      )COPY 1130 GENMETHODS
      ∇JINV[⎕]∇
    ∇ R←JINV M;N;I;K;P;KK;P1;P2;P3;L
[1]   N←(ρM)[1]
[2]   R←((Nρ1),0)\M
[3]   I←1
[4]   R[;N+1]←I=ιN
[5]   P←R[;1]∘.×R[I;]
[6]   K←ιN
[7]   KK←K∘.≠(N+1)ρI
[8]   P1←P×KK
[9]   P2←R×KK
[10]  P3←R×~KK
[11]  L←R[I;1]
[12]  R[;ιN]←(P2-((P1-P3)÷L))[;1+ιN]
[13]  →(N≥I←I+1)/4
[14]  R←((Nρ1),0)/R
    ∇

      ∇DSORT[⎕]∇
    ∇ ORD←DSORT X;INDEX;UNS;WHICH
[1]   ORD←0ρINDEX←ιρUNS←X
[2]   →(0≥ρUNS)/0
[3]   WHICH←UNS=⌈/UNS
[4]   ORD←ORD,WHICH/INDEX
[5]   UNS←(~WHICH)/UNS
[6]   INDEX←(~WHICH)/INDEX
[7]   →2
    ∇

      ∇MEAN[⎕]∇
    ∇ T←MEAN X
[1]   →(1<ρρX)/COL
[2]   T←(+/X)÷ρX
[3]   →0
[4]   COL:T←(+⌿X)÷((ρX)[1])
    ∇

      ∇VAR[⎕]∇
    ∇ R←VAR X
[1]   →(1<ρρX)/COL
[2]   R←(+/(X-MEAN X)*2)÷(ρX)-1
[3]   →0
[4]   COL:R←(+⌿(X-(ρX)ρMEAN X)*2)÷((ρX)[1]-1)
    ∇
```