APL PLUS

STSC PUBLIC LIBRARY 2

SEPTEMBER 1969

STATPAK

STATISTICAL APPLICATIONS

'STATPAK' IS A COLLECTION OF APL FUNCTIONS WHICH PERFORM COMMONLY OCCURRING CALCULATIONS IN STATISTICAL ANALYSIS SUCH AS ANALYSIS OF VARIANCE, REGRESSION AND CORRELATION ANALYSIS, LINEAR PROGRAMMING, AND CRITICAL PATH ANALYSIS.

THIS DOCUMENT WAS PREPARED USING APL-PLUS EDITPAK.

CROSS REFERENCE

FUNCTION	LIB	WORKSPACE	DESCRIPTION
ANOVA	2	STP1	COMPLETE FACTORIAL ANAL. OF VARIANCE
ANOVA1	2	STP1	CROSSED,NESTED FACTORIAL ANAL. OF VAR.
ANOVA2	2	STP1	ONE-WAY ANALYSIS OF VARIANCE
ASSIGN	2	STP4	ASSIGNMENT PROBLEM
BINOM	2	STP3	BINOMIAL PROBABILITIES
CM	2	STP2	SIMPLE CORRELATION MATRIX
CORR	2	STP2	SIMPLE AND PARTIAL CORRELATION
COSTFLOW	2	STP4	MINIMUM COST FLOW
CPM	2	STP4	CRITICAL PATH METHOD
CPM1	2	STP5	CONVERSATIONAL CRITICAL PATH METHOD
CTAB	2	STP3	TWO-WAY CONTINGENCY TABLE
CTRANSPORT	2	STP4	CAPACITATED TRANSPORT PROBLEM
DSTAT	2	STP1	DESCRIPTIVE STATISTICS
FR	2	STP1	ONE-WAY FREQUENCY TABLE
FREQ	2	STP1	NORMAL FIT
FR2	2	STP1	TWO-WAY FREQUENCY TABLE
HIST	2	STP1	FREQUENCY HISTOGRAM
INV	2	STP2	GAUSS-JORDAN MATRIX INVERSION,PIVOTING
JINV	2	STP2	GAUSS-JORDAN MATRIX INVERSION
LINPR	2	STP6	CONVERSATIONAL LINEAR PROGRAMMING
LIST	2	INDEX	LISTS FUNCTIONS OF A GIVEN WORKSPACE
LISTALL	2	INDEX	LISTS FUNCTIONS OF LIB 2 WITH DESC.
LPSOLN	2	STP4	LINEAR PROGRAMMING SENSITIVITY ANAL.
MVSD	2	STP1	MEAN, VARIANCE, AND STANDARD DEVIATION
NETFLOW	2	STP4	FORD-FULKERSON ALGORITHM
NTILES	2	STP1	MEDIAN, QUARTILES, ETC.
PBS	2	STP3	SUM-SCAN OPERATOR
PERMUTE	2	STP3	PERMUTE VECTOR
POISSON	2	STP3	POISSON PROBABILITIES
REG	2	STP2	SIMPLE AND MULTIPLE REGRESSION
RES	2	STP2	RESIDUALS
RND	2	STP3	ROUND
RSIM	2	STP4	REVISED SIMPLEX LINEAR PROGRAMMING
RSIM1	2	STP4	REVISED SIMPLEX LINEAR PROGRAMMING
SCORR	2	STP2	SIMPLE CORRELATIONS
SIMPLEX	2	STP4	SIMPLEX ALGORITHM LINEAR PROGRAMMING
SMOOTH	2	STP3	WEIGHTED MOVING AVERAGE
SORT	2	STP3	NUMERIC SORTING
SR	2	STP2	SIMPLE REGRESSION
STATRES	2	STP2	RESIDUAL STATISTICS
STREG	2	STP2	STEPWISE REGRESSION
TRANSPORT	2	STP4	TRANSPORTATION PROBLEM
UPDATES	2	INDEX	LISTS FNS ENTERED AFTER A GIVEN DATE

TABLE OF CONTENTS

INTRODUCTION

'STATPAK' IS A COLLECTION OF APL FUNCTIONS FOR PERFORMING MANY COMMONLY OCCURRING CALCULATIONS IN STATISTICAL ANALYSIS AND MATHEMATICAL PROGRAMMING. SOME OF THE AREAS COVERED ARE DESCRIPTIVE STATISTICS, FREQUENCY DISTRIBUTION, REGRESSION AND CORRELATION ANALYSIS, ANALYSIS OF VARIANCE, LINEAR PROGRAMMING, AND CRITICAL PATH ANALYSIS. THE FOLLOWING WORKSPACES ARE CURRENTLY INCLUDED IN THE LIBRARY:

WORKSPACE	DESCRIPTION
INDEX	INDEX OF LIBRARY 2 FUNCTIONS.
STP1	DESCRIPTIVE STATISTICS AND ANALYSIS OF VARIANCE.
STP2	REGRESSION AND CORRELATION ANALYSIS.
STP3	PROBABILITY DISTRIBUTIONS AND GENERAL PURPOSE FUNCTIONS.
STP4	OPERATIONS RESEARCH PROGRAMS, INCLUDING ASSIGNMENT, MINIMUM COST FLOW, TRANSPORTATION, LINEAR PROGRAMMING, AND LINEAR PROGRAMMING SENSITIVITY ANALYSIS.
STP5	CRITICAL PATH CALCULATIONS.
STP6	CONVERSATIONAL LINEAR PROGRAMMING.

THE FUNCTIONS IN THIS LIBRARY WERE PRODUCED BY THE DEPARTMENT OF COMPUTING SCIENCES AT THE UNIVERSITY OF ALBERTA, CANADA, AND ARE ALSO DESCRIBED IN THAT DEPARTMENT'S PUBLICATION NO. 17, 'STATPACK2: AN APL STATISTICAL PACKAGE', BY KEITH W. SMILLIE.

TWO GROUP STRUCTURES HAVE BEEN USED IN THE ORGANIZATION OF STP1, STP2, STP3 AND STP4. THE PRIMARY GROUP STRUCTURE GROUPS ALL FUNCTIONS AND ALL ABSTRACTS IN A GIVEN WORKSPACE, AS FOR EXAMPLE 'FNSGRP1' AND 'HOWGRP1' FOR STP1. TO PROVIDE MORE SPACE FOR DATA AND FOR EXECUTION OF FUNCTIONS, THE ABSTRACTS MAY BE DROPPED BY DELETING THE GROUP. FOR EXAMPLE:

```
)LOAD 2 STP2
)ERASE HOWGRP2
```

THE SECOND GROUP STRUCTURE GROUPS SUBFUNCTIONS WITH THE CALLING FUNCTION, AS FOR EXAMPLE 'NTILESGRP' WHICH CONTAINS THE CALLING FUNCTION 'NTILES' AND THE SUBFUNCTION 'PBS'. THE COMMAND:

```
)COPY 2 STP1 NTILESGRP
```

COPIES BOTH 'NTILES' AND 'PBS' FROM 2 STP1.

TO FIND THE GROUP STRUCTURE IN A WORKSPACE, LOAD THAT WORKSPACE AND THEN TYPE:

```
)GRPS
```

A LIST OF GROUP NAMES WILL BE PRINTED. TO FIND THE CONTENTS OF AN INDIVIDUAL GROUP, AS FOR EXAMPLE GROUP 'FNSGRP3', TYPE:

```
)GRP FNSGRP3
```

1. *INDEX*

THIS WORKSPACE IS AN INDEX OF FUNCTIONS STORED IN LIBRARY 2. *THE FOLLOWING FUNCTIONS ARE OF INTEREST TO THE USER:*

SYNTAX	*DESCRIPTION*
LIST	*PRODUCES AN ALPHABETIC LIST OF ALL INDEXED FUNCTIONS STORED IN A SPECIFIC WORKSPACE, GIVING CREATION DATE, A BRIEF DESCRIPTION, AND THE NAME OF AN ASSOCIATED DESCRIPTIVE FUNCTION OR VARIABLE CONTAINING ADDITIONAL INFORMATION ABOUT THE FUNCTION.*
LISTALL	*PRODUCES AN ALPHABETIC LIST OF ALL INDEXED FUNCTIONS IN THIS LIBRARY.*
UPDATES	*PRODUCES AN ALPHABETIC LIST OF ALL ADDITIONS AND CHANGES MADE TO THIS LIBRARY ON OR AFTER A GIVEN DATE.*

2. *STP1*

THIS WORKSPACE CONTAINS THE FOLLOWING FUNCTIONS:

NAME	*DESCRIPTION*
ANOVA	*COMPLETE FACTORIAL ANALYSIS OF VARIANCE. (ANOVAHOW)*
*ANOVA*1	*CROSSED OR NESTED DESIGN ANALYSIS OF VARIANCE. (ANOVA*1*HOW)*
*ANOVA*2	*ONE-WAY ANALYSIS OF VARIANCE. (ANOVA*2*HOW)*
DSTAT	*DESCRIPTIVE STATISTICS OF UNGROUPED OBSERVATIONS. (DSTATHOW)*
FR	*ONE-WAY FREQUENCY TABLE. (FRHOW)*
FREQ	*NORMAL FIT OF GROUPED OBSERVATIONS. (FREQHOW)*
*FR*2	*TWO-WAY FREQUENCY TABLE. (FR*2*HOW)*
HIST	*FREQUENCY HISTOGRAM. (HISTHOW)*
MVSD	*MEAN, VARIANCE, AND STANDARD DEVIATION. (MVSDHOW)*
NTILES	*MEDIAN, QUARTILES, ETC. (NTILESHOW)*

ANOVAHOW

COMPLETE FACTORIAL ANALYSIS OF VARIANCE
T←ANOVA D

THIS PROGRAM DOES AN ANALYSIS OF VARIANCE ON A COMPLETE FACTORIAL DESIGN WITH ARBITRARY NUMBERS OF REPLICATIONS AND FACTORS, WITH ARBITRARY NUMBERS OF LEVELS, WITH THE RESTRICTION THAT THERE ARE NO MISSING OBSERVATIONS.

'T' IS A MATRIX WITH 4 COLUMNS FOR IDENTIFICATION, DEGREES OF FREEDOM, SUMS OF SQUARES, AND MEAN SQUARES. THE ROWS OF 'T' REPRESENT REPLICATIONS, MAIN EFFECTS AND INTERACTIONS, ERROR AND TOTAL.

AS AN EXAMPLE, CONSIDER THE FOLLOWING 2×2 DESIGN WITH 3 REPLICATIONS:

```
1  2     5  6      9  10
3  4     7  8     11  12
```

WHERE THE COLUMNS WITHIN EACH SQUARE REFER TO THE FIRST FACTOR 'A' AND THE ROWS TO THE SECOND FACTOR 'B'. THE DATA SHOULD BE PREPARED AS A VECTOR 1,2,...,12, AND THEN RESTRUCTURED INTO 'D' WITH DIMENSIONS (3,2,2). 'T' WILL HAVE 6 ROWS FOR REPLICATIONS A,B AND AB EFFECTS, ERROR AND TOTAL. THE IDENTIFICATION IN THE FIRST COLUMN WILL BE 1,10 AND 11 FOR A,B AND AB, RESPECTIVELY, AND 0'S FOR THE REMAINING ROWS. IF IT IS DESIRED TO TREAT THE DESIGN AS A 3×2×2 FACTORIAL WITH A SINGLE REPLICATION, THEN 'D' MUST BE RESTRUCTURED TO HAVE DIMENSIONS (1,3,2,2). 'T' WILL THEN HAVE 8 ROWS FOR A,B,C, AB,AC,BC, ABC AND TOTAL. THE ROWS FOR REPLICATIONS AND ERROR WILL BE OMITTED.

'ANOVA' USES THE SUBFUNCTION 'SS'.

ANOVA1HOW

CROSSED OR NESTED DESIGN ANALYSIS OF VARIANCE
ANOVA1

THIS FUNCTION ANALYZES A FACTORIAL DESIGN WITH NO MISSING DATA AS A CROSSED, NESTED OR CROSSED-NESTED DESIGN. REPLICATIONS ARE CONSIDERED AS A FACTOR. THERE MAY BE ANY NUMBER ≥2 OF LEVELS OF ANY NUMBER ≥2 OF FACTORS.

IF THE DATA HAVE NOT BEEN STORED AS A MULTIDIMENSIONAL ARRAY (AS IN 'ANOVA') IN THE GLOBAL VARIABLE 'X', THEN ENTER A VECTOR GIVING THE NUMBER OF LEVELS OF EACH FACTOR (INCLUDING REPLICATIONS) AS THE PARAMETERS. THEN ENTER OBSERVATIONS, ONE AT A TIME, IN THE FORMAT: LEVEL OF 1ST FACTOR, LEVEL OF 2ND FACTOR, ..., OBSERVATION. AFTER THE LAST OBSERVATION HAS BEEN ENTERED, ENTER 0.

AFTER THE GRAND MEAN AND TOTAL DF AND SS HAVE BEEN TYPED OUT, ENTER EFFECTS, ONE AT A TIME, WITH 1-FACTOR EFFECTS FIRST, 2-FACTOR EFFECTS SECOND, AND SO ON. EFFECT 1, FOLLOWED BY EFFECT 2, FOLLOWED BY EFFECT 1 2 GIVES MAIN EFFECTS FOR FACTORS 1 AND 2 AND THEIR INTERACTION. EFFECT 1 FOLLOWED BY EFFECT 1 2 WILL GIVE MAIN EFFECT FOR FACTOR 1 AND THEN 2ND FACTOR NESTED WITHIN 1ST FACTOR. AN EFFECT OF 0 PRODUCES ANY RESIDUAL TERM.

THE DATA 'X' AND RESIDUALS 'RX' ARE GLOBAL VARIABLES.

ANOVA2HOW

ONE-WAY ANALYSIS OF VARIANCE
R←ANOVA2 D

'ANOVA2' PERFORMS AN ANALYSIS OF VARIANCE ON A ONE-WAY CLASSIFICATION WITH MISSING OBSERVATIONS. 'D' IS AN M×N MATRIX, WHERE 'M' IS THE NUMBER OF OBSERVATIONS FOR THE TREATMENT WITH THE MAXIMUM NUMBER OF OBSERVATIONS, AND 'N' IS THE NUMBER OF TREATMENTS. LEGITIMATE OBSERVATIONS ARE GIVEN BY POSITIVE COMPONENTS IN 'D', AND MISSING OBSERVATIONS BY ZERO COMPONENTS. 'R' IS AN (N+3)×4 MATRIX MADE UP AS FOLLOWS:

ROW I:	(I=1,...,N) I NO. OF OBSERVATIONS FOR TREATMENT I, MEAN OF TREATMENT I, 0
ROW N+1:	DF, SS, MS, AND F-RATIO FOR TREATMENTS
ROW N+2:	DF, SS, AND MS FOR ERROR, 0
ROW N+3:	DF, AND SS FOR TOTAL, SQUARE ROOT OF ERROR MS, 0

DSTATHOW

DESCRIPTIVE STATISTICS
DSTAT X

FOR A VECTOR 'X' OF UNGROUPED OBSERVATIONS, DSTAT COMPUTES AND LISTS WITH APPROPRIATE LABELS THE FOLLOWING STATISTICS: SAMPLE SIZE, MAXIMUM OBSERVATION, MINIMUM OBSERVATION, RANGE, MEAN, VARIANCE, STANDARD DEVIATION, MEAN DEVIATION, MEDIAN AND MODE. IF THE MODE OCCURS FOR SEVERAL VALUES, EACH MODE IS LISTED, EXCEPT IF ALL OBSERVATIONS ARE DIFFERENT IN WHICH CASE NO MODE IS LISTED.

FRHOW

ONE-WAY FREQUENCY TABLE
F←P FR X

'F' IS A VECTOR OF FREQUENCIES RESULTING FROM CLASSIFYING THE VECTOR 'X' OF OBSERVATIONS ACCORDING TO THE VECTOR 'P', WHERE P[1] IS THE LEFT-HAND END OF THE FIRST FREQUENCY CLASS, P[2] IS THE CLASS WIDTH, AND P[3] IS THE NUMBER OF CLASSES.

FREQHOW

NORMAL FIT
T←P FREQ X

'T' IS A FREQUENCY TABLE RESULTING FROM CLASSIFYING THE VECTOR 'X' OF OBSERVATIONS ACCORDING TO THE VECTOR 'P', WHERE P[1] IS THE LEFT-HAND END OF THE FIRST FREQUENCY CLASS, P[2] IS THE CLASS WIDTH, P[3] IS THE NUMBER OF CLASSES, AND P[4] IS ANY NUMBER, SAY 0. 'T' IS A MATRIX OF P[3]+2 ROWS AND 6 COLUMNS MADE UP AS FOLLOWS:

COL 1: 0, LEFT-HAND ENDS OF FREQUENCY CLASSES, 0
COL 2: 0, RIGHT-HAND ENDS OF FREQUENCY CLASSES, 0
COL 3: 0, MID-POINTS OF FREQUENCY CLASSES, 0
COL 4: 0, OBSERVED FREQUENCIES, 0
COL 5: EXPECTED FREQUENCIES (INCLUDING TWO TAILS OF DISTRIBUTION)
COL 6: DIFFERENCE BETWEEN OBSERVED AND EXPECTED FREQUENCIES

IF THE LAST COMPONENT OF 'P' IS OMITTED, THE NORMAL FIT IS NOT DONE, AND THUS THE FIRST AND LAST ROWS AND THE LAST TWO COLUMNS OF 'T' ARE OMITTED.

FR2HOW

TWO-WAY FREQUENCY TABLE
T←P FR2 M

'M' IS A MATRIX WITH 2 ROWS GIVING THE (SAME NUMBER OF) OBSERVATIONS ON TWO VARIABLES TO BE CLASSIFIED IN A TWO-WAY FREQUENCY TABLE. 'P' IS A MATRIX WITH 2 ROWS AND 3 COLUMNS WHICH GIVES THE PARAMETERS FOR THE TABLE. THE COMPONENTS IN THE FIRST ROW OF 'P' ARE AS FOLLOWS: P[1;1] IS THE LEFT-HAND END OF THE FIRST FREQUENCY CLASS, P[1;2] IS THE CLASS WIDTH, AND P[1;3] IS THE NUMBER OF CLASSES FOR THE OBSERVATIONS IN THE FIRST ROW OF 'M'. THE COMPONENTS IN THE SECOND ROW OF 'P' ARE SIMILARLY DEFINED FOR THE OBSERVATIONS IN THE SECOND ROW OF 'M'. 'T' IS A MATRIX GIVING THE FREQUENCY TABLE, WHERE THE ROWS CORRESPOND TO THE FIRST VARIABLE AND THE COLUMNS TO THE SECOND VARIABLE.

HISTHOW

FREQUENCY HISTOGRAM
G←W HIST F

'G' IS A FREQUENCY HISTOGRAM GENERATED BY THE VECTOR 'F' OF FREQUENCIES. EACH COMPONENT OF 'F' IS DIVIDED BY THE INTEGER 'W' (W>0) AND ROUNDED BEFORE PLOTTING.

MVSDHOW

MEAN, VARIANCE, AND STANDARD DEVIATION
T←MVSD X

'X' IS A MATRIX WITH THE ROWS CORRESPONDING TO OBSERVATIONS AND THE COLUMNS TO VARIATES. 'T' IS A MATRIX WITH 3 COLUMNS, AND THE NUMBER OF ROWS EQUAL TO THE NUMBER OF COLUMNS OF 'X'. THE FIRST COLUMN OF 'T' GIVES THE MEANS OF THE VARIATES, THE SECOND THE VARIANCES, AND THE THIRD THE STANDARD DEVIATIONS.

NTILESHOW

MEDIAN, QUARTILES, ETC.
T←P NTILES F

'F' IS A VECTOR OF FREQUENCIES. 'P' IS A 3-COMPONENT VECTOR, WHERE P[1] IS THE LEFT-HAND END OF THE FIRST FREQUENCY CLASS, P[2] IS THE CLASS WIDTH, AND P[3] IS AN INTEGER GREATER THAN 0. 'T' IS A VECTOR. IF P[3]=2, 'T' GIVES THE MEDIAN; IF P[3]=4, 'T' GIVES THE 3 QUARTILES; ETC.

'NTILES' USES THE SUBFUNCTION 'PBS'.

3. *STP2*

THIS WORKSPACE CONTAINS THE FOLLOWING FUNCTIONS:

NAME	*DESCRIPTION*
CM	*SIMPLE CORRELATION MATRIX. (CMHOW)*
CORR	*SIMPLE AND PARTIAL CORRELATION. (CORRHOW)*
INV	*GAUSS-JORDAN MATRIX INVERSION WITH PIVOTING. (INVHOW)*
JINV	*GAUSS-JORDAN MATRIX INVERSION. (JINVHOW)*
REG	*SIMPLE AND MULTIPLE REGRESSION. (REGHOW)*
RES	*CALCULATION OF RESIDUALS. (RESHOW)*
SCORR	*CALCULATION OF SIMPLE CORRELATION COEFFICIENTS. (SCORRHOW)*
SR	*SIMPLE REGRESSION ANALYSIS. (SRHOW)*
STATRES	*RESIDUAL STATISTICS. (STATRESHOW)*
STREG	*STEPWISE REGRESSION ANALYSIS. (STREGHOW)*

CMHOW

SIMPLE CORRELATION MATRIX
R←CM X

'R' IS THE MATRIX OF SIMPLE CORRELATION COEFFICIENTS GENERATED FROM THE MATRIX 'X', WHERE THE ROWS OF 'X' CORRESPOND TO OBSERVATIONS AND THE COLUMNS TO VARIATES. FOR EXAMPLE, 10 OBSERVATIONS ON EACH OF 6 VARIATES WOULD BE ASSEMBLED INTO A MATRIX WITH 10 ROWS AND 6 COLUMNS. 'R' WOULD HAVE 6 ROWS AND 6 COLUMNS. THE ELEMENT IN ROW 3 AND COLUMN 5, SAY, OF 'R' IS THE SIMPLE CORRELATION COEFFICIENT BETWEEN VARIATE 3 AND VARIATE 5.

CORRHOW

SIMPLE AND PARTIAL CORRELATION
C←V CORR M

'M' IS THE SQUARE MATRIX OF SIMPLE CORRELATION COEFFICIENTS GIVEN BY THE RESULT 'R' OF THE FUNCTION 'CM'. 'V' IS A VECTOR SPECIFYING THE CORRELATION COEFFICIENT TO BE CALCULATED. FOR EXAMPLE, SUPPOSE 'M' IS OF ORDER 8. THEN IF V=(3,6), 'C' IS THE SIMPLE CORRELATION COEFFICIENT BETWEEN VARIABLES 3 AND 6; IF V=(5,1,8,2), 'C' IS THE PARTIAL CORRELATION COEFFICIENT BETWEEN VARIABLES 5 AND 1 WITH THE EFFECTS OF VARIABLES 8 AND 2 REMOVED.

'CORR' USES THE SUBFUNCTION 'INV'.

INVHOW

GAUSS-JORDAN MATRIX INVERSION WITH PIVOTING
RB←INV RA

'RB' IS THE INVERSE OF THE NON-SINGULAR SQUARE MATRIX 'RA' CALCULATED BY THE GAUSS-JORDAN METHOD WITH PIVOTING FOR MATRIX INVERSION.

JINVHOW

GAUSS-JORDAN MATRIX INVERSION
R←JINV M

'R' IS THE INVERSE OF THE NON-SINGULAR SQUARE MATRIX 'M' CALCULATED BY THE GAUSS-JORDAN METHOD OF MATRIX INVERSION.

REGHOW

SIMPLE AND MULTIPLE REGRESSION
T←V REG X

'X' IS A MATRIX OF OBSERVATIONS, WHERE THE COLUMNS CORRESPOND TO VARIATES AND THE ROWS TO OBSERVATIONS. 'V' IS A VECTOR OF POSITIVE INTEGERS. 'T' IS A MATRIX OF 5 COLUMNS. AS AN EXAMPLE OF THE OUTPUT, LET 'X' HAVE 6 COLUMNS, AND LET V=(3,5,1,4). THEN 'T' GIVES THE RESULTS OF THE BEST LEAST-SQUARES FIT OF THE FUNCTION:

$$X4=A+B\times X3+C\times X5+D\times X1$$

IN THE FOLLOWING FORMAT:

ROW 1:	4, *A*, 0, 0, 0
ROW 2:	3, *B*, *ST ERROR OF B*, *T-VALUE*, 0
ROW 3:	5, *C*, *ST ERROR OF C*, *T-VALUE*, 0
ROW 4:	1, *D*, *ST ERROR OF D*, *T-VALUE*, 0
ROW 5:	0, *DF FOR REGRESSION*, *SUM OF SQUARES*, *MEAN SQUARE*, *F-VALUE*
ROW 6:	0, *DF FOR ERROR*, *SUM OF SQUARES*, *MEAN SQUARE*, 0
ROW 7:	0, *DF FOR TOTAL*, *SUM OF SQUARES*, *ST ERROR OF ESTIMATE*, *SQUARE OF MULTIPLE CORRELATION COEFFICIENT*

'REG' USES THE SUBFUNCTION 'INV'.

RESHOW

RESIDUALS
R←T RES X

'X' IS THE MATRIX OF OBSERVATIONS DEFINED FOR 'REG', AND 'T' IS THE RESULT OF USING 'REG' WITH SOME VECTOR 'V'. 'R' IS A MATRIX WITH 4 COLUMNS AND THE NUMBER OF ROWS EQUAL TO THE NUMBER OF ROWS IN 'X', WHICH GIVES THE FOLLOWING RESULTS OF FITTING THE REGRESSION SPECIFIED BY 'X' AND 'V':

COL 1:	1,2, ...
COL 2:	*OBSERVED VALUES OF DEPENDENT VARIABLE*
COL 3:	*ESTIMATED VALUES OF DEPENDENT VARIABLE*
COL 4:	*RESIDUALS*

SCORRHOW

SIMPLE CORRELATIONS
R←SCORR D

'D' IS A MATRIX OF OBSERVATIONS WITH THE ROWS CORRESPONDING TO OBSERVATIONS AND THE COLUMNS TO VARIATES. MISSING OBSERVATIONS ARE RECORDED IN 'D' AS ANY NEGATIVE NUMBER. THE SIMPLE CORRELATION COEFFICIENT IS COMPUTED BETWEEN EACH DISTINCT PAIR OF VARIATES FOR ALL OBSERVATIONS EXCEPT THOSE IN WHICH EITHER OR BOTH OBSERVATIONS ARE MISSING FOR THE PARTICULAR PAIR OF VARIATES IN QUESTION. 'R' IS A MATRIX WITH 4 COLUMNS IN THE FOLLOWING FORMAT:

COL 1: I=COLUMN INDEX OF FIRST VARIATE
COL 2: J=COLUMN INDEX OF SECOND VARIATE
COL 3: NUMBER OF OBSERVATIONS FOR VARIATES I AND J
COL 4: CORRELATION COEFFICIENT FOR VARIATES I AND J

IF 'D' HAS 'K' COLUMNS, THEN 'R' HAS K(K-1)÷2 ROWS.

'SCORR' USES THE SUBFUNCTION 'PBS'.

SRHOW

SIMPLE REGRESSION
T←X SR Y

'X' AND 'Y' ARE VECTORS GIVING THE (SAME NUMBER OF) OBSERVATIONS ON AN INDEPENDENT VARIABLE 'X' AND A DEPENDENT VARIABLE 'Y'. 'T' IS A MATRIX WITH 5 ROWS AND 3 COLUMNS CONTAINING THE RESULTS OF FITTING THE STRAIGHT LINE Y=A+B×X BY THE METHOD OF LEAST SQUARES. 'T' HAS THE FORMAT:

ROW 1: MEAN OF X, ST DEV OF X, 0
ROW 2: MEAN OF Y, ST DEV OF Y, 0
ROW 3: A, 0, 0
ROW 4: B, ST ERROR OF B, T-VALUE
ROW 5: ST ERROR OF ESTIMATE, R=SIMPLE CORRELATION COEFFICIENT, R*2

STATRESHOW

RESIDUAL STATISTICS
STATRES T

'T' IS THE MATRIX WITH 4 COLUMNS RESULTING FROM THE USE OF THE RESIDUAL FUNCTION 'RES'. ITS LAST COLUMN GIVES THE RESIDUALS RESULTING FROM THE SPECIFIED REGRESSION. THIS FUNCTION GIVES THE FOLLOWING STATISTICS COMPUTED FROM THE RESIDUALS, WITH SUITABLE LABELS:

SUM OF RESIDUALS
SUM OF SQUARES OF RESIDUALS
DURBIN-WATSON STATISTICS
NUMBER OF RUNS OF POSITIVE SIGNS
NUMBER OF RUNS OF NEGATIVE SIGNS
NUMBER OF POSITIVE SIGNS
NUMBER OF NEGATIVE SIGNS
MEAN
STANDARD DEVIATION

THE MEAN AND STANDARD DEVIATION OF THE NUMBER OF RUNS IS BASED ON A NORMAL DISTRIBUTION.

STREGHOW

STEPWISE REGRESSION
T←V STREG X

THIS PROGRAM IS IDENTICAL IN FUNCTION TO THE SIMPLE AND MULTIPLE REGRESSION PROGRAM 'REG' EXCEPT THAT THE INDEPENDENT VARIABLES ARE ENTERED INTO THE REGRESSION IN THE STEPWISE ORDER.

THE VECTOR 'V' IS IDENTICAL TO THE VECTOR 'V' IN 'REG' EXCEPT THAT THE INDEPENDENT VARIABLES MAY BE SPECIFIED IN 'V' IN ANY ORDER. THE FORMAT OF THE MATRIX OF RESULTS 'T' IS IDENTICAL TO THE MATRIX 'T' OF 'REG' EXCEPT THAT THE PROPORTION OF THE VARIATION OF THE DEPENDENT VARIABLE ACCOUNTED FOR BY EACH INDEPENDENT VARIABLE IS GIVEN IN THE FIFTH COLUMN OF 'T' IN THE ROWS CONTAINING THE REGRESSION COEFFICIENTS, STANDARD ERRORS AND T-VALUES FOR THE INDEPENDENT VARIABLES.

'STREG' USES THE SUBFUNCTIONS 'REG', 'INV', AND 'CH'.

4. STP3

THIS WORKSPACE CONTAINS THE FOLLOWING FUNCTIONS:

NAME	DESCRIPTION
BINOM	BINOMIAL PROBABILITIES (BINOMHOW)
CTAB	TWO-WAY CONTINGENCY TABLE (CTABHOW)
PBS	SUM-SCAN OPERATOR (PBSHOW)
PERMUTE	PERMUTE VECTOR (PERMUTEHOW)
POISSON	POISSON PROBABILITIES (POISSONHOW)
RND	ROUNDING (RNDHOW)
SMOOTH	WEIGHTED MOVING AVERAGE (SMOOTHHOW)
SORT	NUMERIC SORTING (SORTHOW)

BINOMHOW

BINOMIAL PROBABILITIES
B←N BINOM P

CALCULATES THE VECTOR 'B' OF PROBABILITIES IN 'N' BINOMIAL TRIALS WITH PROBABILITY 'P' OF SUCCESS IN A SINGLE TRIAL. 'N' AND 'P' ARE SCALARS; 'N' IS A POSITIVE INTEGER, AND 'P' IS BETWEEN 0 AND 1.

CTABHOW

TWO-WAY CONTINGENCY TABLE
R←CTAB T

CALCUALTES CHI-SQUARE AND DEGREES OF FREEDOM FOR A TWO-WAY CONTINGENCY TABLE. THE MATRIX 'T' IS THE CONTINGENCY TABLE. R[1] IS THE DEGREES OF FREEDOM, AND R[2] IS CHI-SQUARE ROUNDED TO 2 DECIMAL PLACES.

'CTAB' USES THE SUBFUNCTION 'RND'.

PBSHOW

SUM-SCAN OPERATOR
V←PBS S

'V' IS THE SUM-SCAN OF THE VECTOR 'S'.

PERMUTEHOW

PERMUTE VECTOR
R←PERMUTE V

'V' AND 'R' ARE VECTORS GIVING TWO DIFFERENT PERMUTATIONS OF THE K≥2 INTEGERS 0,1,2,...,K-1. 'R' IS THE PERMUTATION OBTAINED FROM 'V' BY CONSIDERING 'V' AS A K-DIGIT NUMBER TO THE BASE 'K', AND ADDING (K-1) BASE 'K' SUCCESSIVELY UNTIL THE SUM CONTAINS EACH OF THE DIGITS 0,1,2,...,K-1. EXECUTION OF 'PERMUTE' 'K' FACTORIAL TIMES STARTING WITH 'V' AS ANY PERMUTATION GIVES ALL 'K' FACTORIAL PERMUTATIONS OF 0,1,2,...,K-1.

POISSONHOW

POISSON PROBABILITIES
P←N POISSON K

CALCULATES THE VECTOR 'P' OF THE FIRST N+1 PROBABILITIES FOR A POISSON DISTRIBUTION WITH PARAMETER 'K'. BOTH 'N' AND 'K' ARE SCALAR POSITIVE INTEGERS.

RNDHOW

ROUND
R←N RND X

THE FUNCTION ROUNDS 'X' TO 'N' DECIMAL PLACES. 'N' IS A SCALAR POSITIVE INTEGER, 'X' MAY BE OF ANY ORDER. THE RESULT IS PLACED IN 'R', WHICH HAS THE SAME DIMENSIONS AS 'X'.

SMOOTHHOW

WEIGHTED MOVING AVERAGE
V←W SMOOTH X

'X' IS THE VECTOR TO BE SMOOTHED BY A WEIGHTED MOVING AVERAGE USING THE VECTOR OF WEIGHTS 'W'. 'V' IS THE SMOOTHED VECTOR.

SORTHOW

NUMERIC SORTING
V←SORT X

'X' IS A VECTOR OF NUMBERS, WITH OR WITHOUT DUPLICATES, TO BE SORTED IN ASCENDING ORDER. 'V' IS THE VECTOR OF SORTED NUMBERS.

5. *STP4*

THIS WORKSPACE CONTAINS THE FOLLOWING FUNCTIONS:

NAME	*DESCRIPTION*
ASSIGN	*SOLUTION OF MINIMIZATION ASSIGNMENT PROBLEM. (ASSIGNHOW)*
COSTFLOW	*SOLUTION OF MINIMUM COST FLOW PROBLEM. (COSTFLOWHOW)*
CPM	*CRITICAL PATH ANALYSIS. (CPMHOW)*
CTRANSPORT	*SOLUTION OF CAPACITATED TRANSPORTATION PROBLEM. (CTRANSPORTHOW)*
LPSOLN	*LINEAR PROGRAMMING SENSITIVITY ANALYSIS. (LPSOLNHOW)*
NETFLOW	*FORD-FULKERSON SOLUTION OF A CAPACITATED NETWORK. (NETFLOWHOW)*
RSIM	*REVISED SIMPLEX METHOD LINEAR PROGRAMMING. (RSIMHOW)*
RSIM1	*REVISED SIMPLEX METHOD LINEAR PROGRAMMING WITH CONVERSATIONAL INPUT. (RSIM1HOW)*
SIMPLEX	*SIMPLEX METHOD LINEAR PROGRAMMING. (SIMPLEXHOW)*
TRANSPORT	*PRIMAL-DUAL SOLUTION OF TRANSPORTATION PROBLEM. (TRANSPORTHOW)*

ASSIGNHOW

ASSIGNMENT PROBLEM
ASSIGN N

THIS FUNCTION USES THE HUNGARIAN METHOD TO SOLVE AN N×N MINIMIZATION ASSIGNMENT PROBLEM WITH COST MATRIX 'C'. THE PRICES MAY BE ENTERED EITHER ROW-BY-ROW BY EXECUTING THE FUNCTION, OR MAY BE PREVIOUSLY STORED IN THE N×N MATRIX 'C', WHERE 'C' IS A GLOBAL VARIABLE WHICH IS NOT CHANGED BY THE EXECUTION OF THE FUNCTION.

THE OUTPUT CONSISTS OF THE PROBLEM NUMBER AND DATE, THE COST OF THE OPTIMAL ASSIGNMENT, AND AN N×N LOGICAL ASSIGNMENT MATRIX IN WHICH ELEMENT (I,J) IS 1 IF AND ONLY IF ROW 'I' IS ASSIGNED TO COLUMN 'J' IN THE OPTIMAL ASSIGNMENT.

COSTFLOWHOW

MINIMUM COST FLOW
COSTFLOW MAX

THIS FUNCTION FINDS THE MINIMUM COST OF A GIVEN FLOW OF VALUE 'MAX' (A SCALAR) THROUGH A NETWORK. THE NETWORK IS DEFINED BY TWO GLOBAL VARIABLES 'G' AND 'C' WHICH ARE N×N MATRICES. G[I;J] IS THE CAPACITY OF THE ARC FROM NODE 'I' TO NODE 'J', AND C[I;J] IS THE COST OF MOVING ONE UNIT FROM NODE 'I' TO NODE 'J'.

THE FLOW VALUE (WHICH MAY BE LESS THAN 'MAX' IF SUCH A FLOW IS NOT POSSIBLE) AND THE COST OF THIS FLOW ARE GIVEN IN THE GLOBAL SCALAR VARIABLES 'FLOW' AND 'MINCOST', RESPECTIVELY. THE EXCESS ARC CAPACITIES ARE GIVEN BY THE MATRIX 'G'. THE NODE NUMBERS ON THE LAST FLOW-AUGMENTING PATH ARE GIVEN BY THE VECTOR 'PATH'. (IF 'G' AND 'C' ARE NOT SQUARE MATRICES OF THE SAME ORDER, THEN 'PATH' IS SET TO AN EMPTY VECTOR AND COMPUTATION STOPS.) 'MAXFLOW' IS A GLOBAL BINARY SCALAR VARIABLE WHICH IS SET TO 1 WHEN A FLOW OF 'MAX' IS ATTAINED.

CPMNOW

CRITICAL PATH METHOD
CPT←T CPM NTW

'CMP' PERFORMS A CRITICAL PATH ANALYSIS ON THE NETWORK REPRESENTED BY THE MATRIX 'NTW' USING THE NODE-ORIENTED METHOD. 'T' IS A SCALAR WHICH IS 0 (OR 1) ACCORDING AS THE IMMEDIATE PREDECESSORS (OR SUCCESSORS) OF EACH JOB (NODE) ARE SPECIFIED.

INPUT: 'NTW' IS A MATRIX WITH 'N' ROWS AND 'M' COLUMNS, WHERE 'N' IS THE NUMBER OF NODES NUMBERED FROM 1 TO 'N', INCLUSIVE, IN ANY ORDER. COLUMN 1 OF 'NTW' GIVES THE NODE NUMBERS, COLUMN 2 THE NODE TIMES AS INTEGERS AND COLUMNS 3 TO 'M' THE IMMEDIATE PREDECESSORS OR SUCCESSORS OF THE NODES. IF ANY NODE HAS LESS THAN M-2 IMMEDIATE PREDECESSORS OR SUCCESSORS, THE REMAINING COLUMNS IN THE CORRESPONDING ROW MUST BE FILLED OUT WITH ZEROS.

OUTPUT: FOR EACH ANALYSIS THE FOLLOWING OUTPUT IS GIVEN AS A MATRIX WITH ONE ROW FOR EACH NODE: NODE NUMBER IN SUCCESSOR ORDER (THAT IS NODE 'I' IS GIVEN BEFORE THE NODES WHICH ARE ITS IMMEDIATE SUCCESSORS), NODE TIME, EARLIEST START TIME, TOTAL SLACK AND FREE SLACK. EACH ROW ON THE CRITICAL PATH, THAT IS NODES WITH TOTAL SLACK OF ZERO, IS PRECEDED BY AN ASTERISK. THE OUTPUT IS ALSO STORED, APART FROM THE ASTERISKS, IN THE MATRIX 'CPT'.

SECONDARY OUTPUT: THE NETWORK 'NTW' SORTED IN THE SAME NODE ORDER AS 'CPT', IS STORED IN THE MATRIX 'CPA'. THE LOGICAL MATRIX 'PSM' WITH 'N' ROWS AND COLUMNS GIVES THE PREDECESSOR-SUCCESSOR RELATIONSHIPS BETWEEN NODES. ROW 'I' GIVES THE SUCCESSORS OF THE I-TH NODE OF 'CPA' AND COLUMN 'I' GIVES THE PREDECESSORS, WHERE AN ENTRY OF 1 INDICATES A PREDECESSOR-SUCCESSOR RELATIONSHIP.

ERROR EXITS: IF THE 'N' NODES ARE NOT NUMBERED, IN SOME ORDER, FROM 1 TO 'N' INCLUSIVE, 'NODE NUMBERING ERROR' FOLLOWED BY A VECTOR OF SORTED NODE NUMBERS IS GIVEN, AND COMPUTATION IS ENDED. IF THERE ARE ANY LOOPS OR DISCONTINUITIES IN THE NETWORK 'LOOP/DISCONTINUITY AT NODE' FOLLOWED BY THE NODE NUMBERS AT OR PRECEDING WHICH AN ERROR OCCURRED IS GIVEN, AND COMPUTATION IS ENDED.

REFERENCE: LEVY, THOMPSON AND WEST, HARVARD BUSINESS REVIEW, VOL. 41, NO.5, PP98-108, SEPT.-OCT., 1963.

CTRANSPORTHOW

CAPACITATED TRANSPORTATION PROBLEM
S←CAP CTRANSPORT COST

THIS FUNCTION USES THE PRIMAL-DUAL ALGORITHM TO SOLVE THE CAPACITATED TRANSPORTATION PROBLEM. 'COST' IS AN (M+1)×(N+1) MATRIX IN WHICH COST[I;J] IS THE UNIT COST OF SHIPPING FROM ORIGIN 'I' TO DESTINATION 'J', WHERE I=1,2,...,M AND J=1,2,...,N. COST[I;N+1] IS THE AMOUNT AVAILABLE AT ORIGIN I=1,2,...,M AND COST[M+1;J] IS THE AMOUNT REQUIRED AT DESTINATION J=1,2,...,N. THE SUM OF THE ORIGIN AVAILABILITIES MUST BE EQUAL TO THE SUM OF THE DESTINATION REQUIREMENTS. 'CAP' IS AN M×N MATRIX IN WHICH CAP[I;J] IS THE CAPACITY RESTRICTION ON THE ROUTE FROM ORIGIN 'I' TO DESTINATION 'J'.

'S' IS AN M×N MATRIX IN WHICH S[I;J] GIVES THE NUMBER OF UNITS SHIPPED FROM ORIGIN 'I' TO DESTINATION 'J' IN THE OPTIMAL SOLUTION. THE CORRESPONDING MINIMUM COST IS GIVEN IN THE GLOBAL SCALAR VARIABLE 'MINCOST'. IF THERE IS NO FEASIBLE SOLUTION, A SUITABLE MESSAGE IS GIVEN IN 'S' WHICH IS THEN A MATRIX WITH ONE ROW, AND 'MINCOST' IS AN EMPTY VECTOR.

'CTRANSPORT' USES THE SUBFUNCTION 'NETFLOW'.

LPSOLNHOW

LINEAR PROGRAMMING SENSITIVITY ANALYSIS
B LPSOLN A

THIS FUNCTION RECALCULATES THE OPTIMAL SOLUTION AND DOES A SENSITIVITY ANALYSIS FOR THE PRICE AND REQUIREMENT VECTORS FOR A LINEAR PROGRAMMING PROBLEM.

THE MATRIX 'A' IS THE SAME MATRIX 'A' REQUIRED FOR EITHER 'SIMPLEX' OR 'RSIM', AND THE VECTOR 'B' GIVES THE VARIABLES IN THE OPTIMAL BASIS. 'LPSOLN' SHOULD NOT BE USED IF SOME COLUMNS OF 'A' CORRESPOND TO ARTIFICIAL VARIABLES. THE OUTPUT CONSISTS OF THE FOLLOWING INFORMATION WITH IDENTIFYING LABELS:

MAXIMUM VALUE OF OBJECTIVE FUNCTION.
VARIABLES IN OPTIMAL BASIS AND THEIR VALUES.
MARGINAL VALUE, LOWER BOUND, RIGHT-HAND SIDE AND UPPER BOUND
FOR THE RIGHT-HAND SIDE OF EACH CONSTRAINT
LOWER BOUND, PRICE AND UPPER BOUND FOR EACH NON-ZERO PRICE.

INFINITE LOWER AND UPPER BOUNDS ARE INDICATED BY VALUES OF ¯7.237E75 AND 7.237E75, RESPECTIVELY.

NETFLOWHOW

FORD-FULKERSON ALGORITHM FOR A CAPACITATED NETWORK
NETFLOW

THIS FUNCTION USES THE FORD-FULKERSON ALGORITHM TO CALCULATE THE MAXIMUM FLOW IN A CAPACITATED NETWORK. THE INPUT CONSISTS OF TWO GLOBAL VARIABLES 'G' AND 'N', WHERE 'G' IS AN N×N CAPACITY MATRIX IN WHICH G[I;J] IS THE CAPACITY OF THE DIRECTED ARC FROM NODE 'I' TO NODE 'J'. 'N' IS A SCALAR. THE FUNCTION LEAVES THE MAXIMUM FLOW IN THE GLOBAL SCALAR VARIABLE 'FLOW' AND THE EXCESS ARC CAPACITIES IN THE MATRIX 'G'. IT ALSO GENERATES AN N-COMPONENT GLOBAL VECTOR 'DELTA'.

REFERENCE: 'LINEAR PROGRAMMING', BY G. HADLEY, ADDISON-WESLEY, 1962.

RSIMHOW

LINEAR PROGRAMMING
T←RSIM A

THIS FUNCTION USES THE REVISED SIMPLEX ALGORITHM TO SOLVE THE LINEAR PROGRAMMING PROBLEM SPECIFIED BY THE MATRIX 'A'. THE FIRST ROW OF 'A', EXCEPT FOR THE LAST COLUMN, WHICH IS ALWAYS ZERO, GIVES THE PRICES. THE REMAINING ROWS GIVE THE CONSTRAINTS WITH NECESSARY SLACK AND SURPLUS VARIABLES, AND THE REQUIREMENTS VECTOR WITH NON-NEGATIVE COMPONENTS IN THE LAST COLUMN. NOTE THAT ARTIFICIAL VARIABLES ARE NOT REQUIRED.

'T' IS A MATRIX WITH 2 COLUMNS. THE FIRST COLUMN GIVES THE VARIABLES IN THE OPTIMAL BASIS EXCEPT FOR THE LAST ROW WHICH IS ALWAYS ZERO. THE SECOND COLUMN GIVES THE VARIABLES IN THE OPTIMAL BASIS EXCEPT FOR THE LAST ROW WHICH GIVES THE OPTIMAL VALUE OF THE OBJECTIVE FUNCTION. IF THE PROBLEM HAS EITHER AN UNBOUNDED SOLUTION OR A NON-FEASIBLE SOLUTION, THEN THE PROPER INDICATION IS GIVEN IN 'T' WHICH IS THEN AN ALPHANUMERIC MATRIX WITH A SINGLE ROW.

EACH LINEAR PROGRAMMING PROBLEM SOLVED WITH THIS FUNCTION MUST BE A MAXIMIZATION PROBLEM.

RSIM1HOW

LINEAR PROGRAMMING
RSIM1

THIS FUNCTION ASSEMBLES THE INPUT FOR THE FUNCTION 'RSIM', TRANSFERS CONTROL TO IT, AND FORMATS THE OUTPUT WITH SUITABLE LABELS. THE DOCUMENTATION FOR 'RSIM' SHOULD BE CONSULTED FOR THE FORM IN WHICH THE OBJECTIVE FUNCTION AND CONSTRAINTS SHOULD BE PUT.

'RSIM1' USES THE SUBFUNCTION 'RSIM'.

SIMPLEXHOW

LINEAR PROGRAMMING
R←B SIMPLEX A

THIS FUNCTION USES THE SIMPLEX ALGORITHM TO SOLVE THE LINEAR PROGRAMMING PROBLEM SPECIFIED BY THE MATRIX 'A'. THE ROWS OF 'A' GIVE THE CONSTRAINTS WITH NECESSARY SLACK AND SURPLUS VARIABLES, AND ALSO ARTIFICIAL VARIABLES WHERE NECESSARY TO MAKE UP AN IDENTITY MATRIX. THE LAST COLUMN OF 'A', EXCEPT FOR THE FIRST ELEMENT WHICH IS ALWAYS ZERO, GIVES THE REQUIREMENTS VECTOR. THE FIRST ROW OF 'A', EXCEPT FOR THE LAST ELEMENT, GIVES THE PRICES. 'B' IS A VECTOR OF POSITIVE INTEGERS WHICH GIVES THE COLUMN NUMBERS CORRESPONDING TO THE IDENTITY MATRIX.

'R' IS A MATRIX WITH 2 COLUMNS. THE FIRST COLUMN GIVES THE VARIABLES IN THE OPTIMAL BASIS EXCEPT FOR THE LAST ROW WHICH IS ALWAYS ZERO. THE SECOND COLUMN GIVES THE VARIABLES IN THE OPTIMAL BASIS EXCEPT FOR THE LAST ROW WHICH GIVES THE OPTIMAL VALUE OF THE OBJECTIVE FUNCTION. IF THE PROBLEM HAS AN UNBOUNDED SOLUTION, THEN THE PROPER INDICATION IS GIVEN IN 'R' WHICH IS THEN AN ALPHANUMERIC VECTOR.

EACH LINEAR PROGRAMMING PROBLEM SOLVED WITH THIS FUNCTION MUST BE A MAXIMIZING PROBLEM.

TRANSPORTHOW

TRANSPORTATION PROBLEM
S←TRANSPORT COST

THIS FUNCTION USES THE PRIMAL-DUAL ALGORITHM TO SOLVE THE TRANSPORTATION PROBLEM. 'COST' IS AN (M+1)×(N+1) MATRIX IN WHICH COST[I;J] IS THE UNIT COST OF SHIPPING FROM ORIGIN 'I' TO DESTINATION 'J', WHERE I=1,2,...,M AND J=1,2,...,N. COST[I;N+1] IS THE AMOUNT AVAILABLE AT ORIGIN 'I', I=1,2,...,M, AND COST[M+1;J] IS THE AMOUNT REQUIRED AT DESTINATION J, J=1,2,...,N. COST[M+1;N+1] IS ARBITRARY. THE SUM OF THE ORIGIN AVAILABILITIES MUST BE EQUAL TO THE SUM OF THE DESTINATION REQUIREMENTS.

'S' IS AN M×N MATRIX WITH S[I;J] GIVING THE NUMBER OF UNITS SHIPPED FROM ORIGIN 'I' TO DESTINATION 'J' IN THE OPTIMAL SOLUTION. THE CORRESPONDING MINIMUM COST IS GIVEN IN THE GLOBAL SCALAR VARIABLE 'MINCOST'.

'TRANSPORT' USES THE SUBFUNCTION 'NETFLOW'.

6. *STP5*

THIS WORKSPACE CONTAINS FUNCTIONS WHICH PERFORM A CRITICAL PATH ANALYSIS. THE USER ACCESSED FUNCTION IS 'CPM1'. (SEE 'CPM1HOW')

CPM1HOW

CRITICAL PATH METHOD
CPM1

'CPM1' IS A SET OF FUNCTIONS FOR PERFORMING A CRITICAL PATH ANALYSIS FOR AN ACTIVITY-ORIENTED NETWORK. THE INPUT CONSISTS OF THE NODE NUMBER, NODE DURATION AND SUCCESSOR NODE NUMBERS FOR EACH NODE. IF THERE ARE N NODES, THEY SHOULD BE NUMBERED IN ANY ORDER FROM 1 TO N. FOR CONVENIENCE OF OUTPUT THE DURATIONS SHOULD BE INTEGERS. THE OUTPUT CONSISTS OF THE FOLLOWING INFORMATION WITH SUITABLE LABELS: PROBLEM NUMBER AND DATE, LENGTH OF CRITICAL PATH, CRITICAL ACTIVITIES, NODE NUMBERS, DURATIONS, EARLY START AND FINISH TIMES, LATE START AND FINISH TIMES, TOTAL AND FREE SLACKS. ERROR INDICATIONS ARE GIVEN FOR INCORRECT NODE NUMBERING, MULTIPLE INITIAL AND TERMINAL NODES, AND NETWORK LOOPS, AND COMPUTATION IS THEN TERMINATED.

TO EXECUTE THE PROGRAM, TYPE 'CPM1'. IF THE INPUT DATA ARE TO BE TYPED IN, ANSWER 'YES' TO THE QUESTION 'IS THIS A NEW PROBLEM?', AND FOLLOW THE INSTRUCTIONS THAT ARE TYPED OUT. THE INITIAL DATA ARE STORED, ONE ROW FOR EACH NODE, IN THE GLOBAL VARIABLE 'DATA' SO THAT CORRECTIONS MAY BE MADE TO THE DATA WITHOUT ENTERING ALL THE DATA AGAIN. TO RESTART AFTER MAKING CORRECTIONS TO 'DATA', ANSWER 'NO' TO THE QUESTION 'IS THIS A NEW PROBLEM?'.

7. *STP6*

THIS WORKSPACE CONTAINS FUNCTIONS WHICH SOLVE LINEAR PROGRAMMING PROBLEMS USING THE REVISED SIMPLEX ALGORITHM. THE APPROACH IS COMPLETELY CONVERSATIONAL USING THE CUSTOMARY ALGEBRAIC STATEMENT OF THE PROBLEM FOUND IN MOST LINEAR PROGRAMMING TEXTBOOKS. THE USER ACCESSED FUNCTION IS 'LINPR'. (SEE 'LINPRHOW')

LINPRHOW

CONVERSATIONAL LINEAR PROGRAMMING
LINPR

THIS PROGRAM SOLVES A LINEAR PROGRAMMING PROBLEM USING THE REVISED SIMPLEX ALGORITHM. THE PROBLEM IS STATED IN THE CUSTOMARY ALGEBRAIC MANNER FOUND IN MOST TEXTBOOKS ON LINEAR PROGRAMMING. THE PROGRAM CAN ACCOMODATE A MAXIMUM OF 30 VARIABLES (INCLUDING SLACK AND SURPLUS VARIABLES) AND 15 CONSTRAINTS.

LET US CONSIDER THE EXAMPLE ON PAGE 16 OF 'AN INTRODUCTION TO LINEAR PROGRAMMING', IBM DATA PROCESSING APPLICATION E20-8171. THIS PROBLEM SHOULD BE ENTERED IN THE FOLLOWING MANNER:

```
MAXIMIZE
     Z=3X1+2X2+2.5X3
SUBJECT TO
     4X1+2X2+2X3≤12
      X1    + X3≤2
         X2 +3X3≤4
END
```

ANY NUMBER OF SPACES MAY BE LEFT WITHIN EACH LINE, AND ANY NUMBER OF BLANK LINES MAY SEPARATE EACH LINE OF INPUT. HOWEVER, THE ORDER OF INPUT MUST BE THE FOLLOWING:

1) 'MAXIMIZE' OR 'MINIMIZE'
2) OBJECTIVE FUNCTION 'Z= ... '
3) 'SUBJECT TO'
4) CONSTRAINTS, IN ANY ORDER
5) 'END'

THE VARIABLES MUST BE DESIGNATED X1, X2, X3, ... ,XN. ONE OF THE THREE SIGNS ≤, =, ≥ MUST BE USED IN EACH CONSTRAINT, WHOSE RIGHT-HAND SIDE MUST BE A NON-NEGATIVE INTEGER.

SLACK AND SURPLUS VARIABLES ARE ADDED AUTOMATICALLY BY THE PROGRAM TO THE CONSTRAINTS IF THEY DO NOT APPEAR, WHERE NECESSARY, IN THE ALGEBRAIC FORMULATION OF THE PROBLEM.

AFTER 'END' HAS BEEN ENTERED, THE OPTIMAL SOLUTION WILL BE COMPUTED USING THE REVISED SIMPLEX ALGORITHM 'RSIM' AND THE RESULTS TYPED. IF THERE IS EITHER NO FEASIBLE SOLUTION OR AN UNBOUNDED SOLUTION, AN INDICATION OF THIS CONDITION WILL BE GIVEN. THEN 'ANY MORE PROBLEMS?' WILL BE TYPED. AN ANSWER OF 'YES' CAUSES THE PROGRAM TO ACCEPT DATA FOR ANOTHER CASE. AN ANSWER OF 'NO' TERMINATES EXECUTION.

INPUT OF ANY LINE MAY BE TERMINATED BY TYPING '/' AS THE LAST CHARACTER IN THE LINE. THE NEXT LINE TYPED REPLACES THE LINE THAT WAS TERMINATED. INPUT OF ANY PROBLEM MAY BE TERMINATED AT ANY STAGE BY ENTERING 'STOP' AS A SEPARATE LINE OF INPUT. COMPUTATION WILL BE BYPASSED, AND THE PROGRAM WILL THEN TYPE OUT 'ANY MORE PROBLEMS?'.