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

THIS DOCUMENT WAS PREPARED USING APL-PLUS EDITTAK.
### CROSS REFERENCE

<table>
<thead>
<tr>
<th>FUNCTION</th>
<th>LIR WORKSPACE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANOVA</td>
<td>2 STP1</td>
<td>COMPLETE FACTORIAL ANAL. OF VARIANCE</td>
</tr>
<tr>
<td>ANOVA1</td>
<td>2 STP1</td>
<td>CROSSED, NESTED FACTORIAL ANAL. OF VARIANCE</td>
</tr>
<tr>
<td>ANOVA2</td>
<td>2 STP1</td>
<td>ONE-WAY ANALYSIS OF VARIANCE</td>
</tr>
<tr>
<td>ASSIGN</td>
<td>2 STP4</td>
<td>ASSIGNMENT PROBLEM</td>
</tr>
<tr>
<td>BINOM</td>
<td>2 STP3</td>
<td>BINOMIAL PROBABILITIES</td>
</tr>
<tr>
<td>CH</td>
<td>2 STP2</td>
<td>SIMPLE CORRELATION MATRIX</td>
</tr>
<tr>
<td>COHR</td>
<td>2 STP2</td>
<td>SIMPLE AND PARTIAL CORRELATION</td>
</tr>
<tr>
<td>COSTFLO</td>
<td>2 STP4</td>
<td>MINIMUM COST FLOW</td>
</tr>
<tr>
<td>CPE</td>
<td>2 STP4</td>
<td>CRITICAL PATH METHOD</td>
</tr>
<tr>
<td>CPBMW</td>
<td>2 STP5</td>
<td>CONVERSATIONAL CRITICAL PATH METHOD</td>
</tr>
<tr>
<td>CTAB</td>
<td>2 STP3</td>
<td>TWO-WAY CONTINGENCY TABLE</td>
</tr>
<tr>
<td>CTABST</td>
<td>2 STP4</td>
<td>CAPACITATED TRANSPORT PROBLEM</td>
</tr>
<tr>
<td>DENS</td>
<td>2 STP1</td>
<td>DESCRIPTIVE STATISTICS</td>
</tr>
<tr>
<td>FR</td>
<td>2 STP1</td>
<td>ONE-WAY FREQUENCY TABLE</td>
</tr>
<tr>
<td>FREQ</td>
<td>2 STP1</td>
<td>NORMAL FIT</td>
</tr>
<tr>
<td>FR2</td>
<td>2 STP1</td>
<td>TWO-WAY FREQUENCY TABLE</td>
</tr>
<tr>
<td>HIST</td>
<td>2 STP1</td>
<td>FREQUENCY HISTOGRAM</td>
</tr>
<tr>
<td>JRV</td>
<td>2 STP2</td>
<td>GAUSS-JORDAN MATRIX INVERSION, PIVOTING</td>
</tr>
<tr>
<td>JRVV</td>
<td>2 STP2</td>
<td>GAUSS-JORDAN MATRIX INVERSION</td>
</tr>
<tr>
<td>LIMPR</td>
<td>2 STP6</td>
<td>CONVERSATIONAL LINEAR PROGRAMMING</td>
</tr>
<tr>
<td>LIST</td>
<td>2 INDEX</td>
<td>LISTS FUNCTIONS OF A GIVEN WORKSPACE</td>
</tr>
<tr>
<td>LISTS2L</td>
<td>2 INDEX</td>
<td>LISTS FUNCTIONS OF LIR 2 WITH DESC.</td>
</tr>
<tr>
<td>LPSELE</td>
<td>2 STP4</td>
<td>LINEAR PROGRAMMING SENSITIVITY ANAL.</td>
</tr>
<tr>
<td>MVSE</td>
<td>2 STP1</td>
<td>MEAN, VARIANCE, AND STANDARD DEVIATION</td>
</tr>
<tr>
<td>NESTFLO</td>
<td>2 STP3</td>
<td>FORD-PULFRINSON ALGORITHM</td>
</tr>
<tr>
<td>POLY</td>
<td>2 STP1</td>
<td>MEDIAN, QUARTILES, ETC.</td>
</tr>
<tr>
<td>POIS</td>
<td>2 STP2</td>
<td>SUM-SCALE OPERATOR</td>
</tr>
<tr>
<td>PSS</td>
<td>2 STP3</td>
<td>PERMUTE VECTOR</td>
</tr>
<tr>
<td>POISSON</td>
<td>2 STP3</td>
<td>POISSON PROBABILITIES</td>
</tr>
<tr>
<td>REG</td>
<td>2 STP2</td>
<td>SIMPLE AND MULTIPLE REGRESSION</td>
</tr>
<tr>
<td>RES</td>
<td>2 STP2</td>
<td>RESIDUALS</td>
</tr>
<tr>
<td>RND</td>
<td>2 STP3</td>
<td>ROUND</td>
</tr>
<tr>
<td>RSIM</td>
<td>2 STP4</td>
<td>REVISIED SIMPLEX LINEAR PROGRAMMING</td>
</tr>
<tr>
<td>RSIM2</td>
<td>2 STP4</td>
<td>REVISIED SIMPLEX LINEAR PROGRAMMING</td>
</tr>
<tr>
<td>SCORR</td>
<td>2 STP2</td>
<td>SIMPLE CORRELATIONS</td>
</tr>
<tr>
<td>SIMPLEX</td>
<td>2 STP4</td>
<td>SIMPLEX ALGORITHM LINEAR PROGRAMMING</td>
</tr>
<tr>
<td>solution</td>
<td>2 STP3</td>
<td>WEIGHTED MOVING AVERAGE</td>
</tr>
<tr>
<td>SORT</td>
<td>2 STP3</td>
<td>NUMERIC SORTING</td>
</tr>
<tr>
<td>SR</td>
<td>2 STP2</td>
<td>SIMPLE REGRESSION</td>
</tr>
<tr>
<td>STAT'S</td>
<td>2 STP2</td>
<td>RESIDUAL STATISTICS</td>
</tr>
<tr>
<td>STREG</td>
<td>2 STP2</td>
<td>STEPWISE REGRESSION</td>
</tr>
<tr>
<td>TRANSP</td>
<td>2 STP8</td>
<td>TRANSPORTATION PROBLEM</td>
</tr>
<tr>
<td>UPDATES</td>
<td>2 INDEX</td>
<td>LISTS PHS ENTERED AFTER A GIVEN DATE</td>
</tr>
</tbody>
</table>
# Table of Contents

**Introduction** ........................................... 3

**Workspace Documentation** ................................

1. **Index** .................................................. 5

2. **STP1** .................................................... 6
   - ANOVA .................................................. 7
   - ANOVA2 .................................................. 8
   - ANOVA3 .................................................. 8
   - BSTAT .................................................. 9
   - FR ....................................................... 9
   - FREQ .................................................... 9
   - FRQ ...................................................... 10
   - HIST .................................................... 10
   - MWSD .................................................... 10
   - NVALUES ................................................ 11

3. **STP2** .................................................... 12
   - CH ....................................................... 13
   - CORR ..................................................... 13
   - IREV ..................................................... 13
   - JREV ..................................................... 13
   - REG ....................................................... 14
   - RES ....................................................... 14
   - SCORE ..................................................... 15
   - SE ......................................................... 15
   - STATUS ................................................... 16
   - STRREV ................................................... 16

4. **STP3** .................................................... 17
   - BJON ..................................................... 18
   - CTAB ...................................................... 18
   - PES ....................................................... 18
   - PERMUT ................................................... 18
   - POISSON .................................................. 18
   - RND ....................................................... 19
   - SMOOTH ................................................... 19
   - SORT ..................................................... 19

5. **STP4** .................................................... 20
   - ASSIGN .................................................. 21
   - COSTFLOW ............................................... 21
   - CPS ....................................................... 22
   - CTRANSPORT ............................................ 23
   - LCONLN .................................................. 23
   - NETFLOW ................................................ 24
   - RESM ..................................................... 24
| RSIM1          | ............................................... | 25 |
| SIMPLEX       | ............................................... | 25 |
| TRANSPORT     | ............................................... | 26 |
| 6. STP5       | ............................................... | 27 |
| CPM1          | ............................................... | 27 |
| 7. STP6       | ............................................... | 28 |
| LINTR         | ............................................... | 29 |
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:

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

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, 'STATPAK2: 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 'Проcgrp1' AND 'Проcgrp1' FOR STP1. TO PROVIDE MORE SPACE FOR DATA AND FOR EXECUTION OF FUNCTIONS, THE ABSTRACTS MAY BE DROPPED BY DELETING THE GROUP. FOR EXAMPLE:

```apl
)load 2 stp2
)erase group1
```
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:**

<table>
<thead>
<tr>
<th>NAME</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANOVA</td>
<td>COMPLETE FACTORIAL ANALYSIS OF VARIANCE. (ANOVAHOW)</td>
</tr>
<tr>
<td>ANOVA1</td>
<td>CROSSED OR NESTED DESIGN ANALYSIS OF VARIANCE. (ANOVA1HOW)</td>
</tr>
<tr>
<td>ANOVA2</td>
<td>ONE-WAY ANALYSIS OF VARIANCE. (ANOVA2HOW)</td>
</tr>
<tr>
<td>DSTAT</td>
<td>DESCRIPTIVE STATISTICS OF UNGROUPED OBSERVATIONS. (DSTATHOW)</td>
</tr>
<tr>
<td>FR</td>
<td>ONE-WAY FREQUENCY TABLE. (FRHOW)</td>
</tr>
<tr>
<td>FREQ</td>
<td>NORMAL FIT OF GROUPED OBSERVATIONS. (FREQHOW)</td>
</tr>
<tr>
<td>FR2</td>
<td>TWO-WAY FREQUENCY TABLE. (FR2HOW)</td>
</tr>
<tr>
<td>HIST</td>
<td>FREQUENCY HISTOGRAM. (HISTHOW)</td>
</tr>
<tr>
<td>MVSD</td>
<td>MEAN, VARIANCE, AND STANDARD DEVIATION. (MVSDHOW)</td>
</tr>
<tr>
<td>NTILES</td>
<td>MEDIAN, QUANTILES, ETC. (NTILESHOW)</td>
</tr>
</tbody>
</table>
**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 2x2 DESIGN WITH 3 REPlications:**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>7</td>
<td>8</td>
<td>11</td>
</tr>
</tbody>
</table>

**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 5 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 3x2x2 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'.

---

**Notes:**

- The text appears to be a manual or documentation for a statistical analysis program, specifically for conducting ANOVA (Analysis of Variance) on complete factorial designs.

- The program can handle arbitrary numbers of replications and factors, with no restrictions on the number of levels.

- A specific example is given for a 2x2 design with 3 replications, illustrating how data should be structured and how the program processes it.

- The program uses matrices to organize the data, with columns for identification, degrees of freedom, sums of squares, and mean squares.

- The output includes rows for main effects, interactions, error, and total, with specific rows dedicated to different effects.

- The program's flexibility is highlighted, allowing for different design structures depending on the user's needs.
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 'ANOVA1') 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 THE 2ND FACTOR TESTED WITHIN 1ST FACTOR. AN EFFECT OF C PRODUCES ANY RESIDUAL TERM.

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

ANOVA2HOW

ONE-WAY ANALYSIS OF VARIANCE

R=ANOVA2 V

'ANOVA2' PERFORMS AN ANALYSIS OF VARIANCE ON A ONE-WAY CLASSIFICATION WITH MISSING OBSERVATIONS. 'D' IS AN MxN 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. 'K' IS AN (K+3)xN MATRIX MADE UP AS FOLLOWS:

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

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.

FREQU

ONE-WAY FREQUENCY TABLE
P=D IN X

'P' IS A VECTOR OF FREQUENCIES RESULTING FROM CLASSIFYING THE
VECTOR 'X' OF OBSERVATIONS ACCORDING TO THE VECTOR 'P', WHERE

FREQU

- NORMAL FIT
T=O P IN X

'T' IS A FREQUENCY TABLE RESULTING FROM CLASSIFYING THE VECTOR
'X' OF OBSERVATIONS ACCORDING TO THE VECTOR 'P', WHERE P[1] IS
0. 'T' IS A MATRIX OF P[3]X2 ROWS AND 6 COLUMNS MADE UP AS
FOLLOWS:

<table>
<thead>
<tr>
<th>COL 1</th>
<th>COL 2</th>
<th>COL 3</th>
<th>COL 4</th>
<th>COL 5</th>
<th>COL 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>NO. LEFT-HAND ENDS OF FREQUENCY CLASSES</td>
<td>NO. RIGHT-HAND ENDS OF FREQUENCY CLASSES</td>
<td>NO. MID-POINTS OF FREQUENCY CLASSES</td>
<td>NO. OBSERVED FREQUENCIES</td>
<td>NO. EXPECTED FREQUENCIES (INCLUDING TWO TAILS OF DISTRIBUTION)</td>
<td>NO. DIFFERENCE BETWEEN OBSERVED AND EXPECTED FREQUENCIES</td>
</tr>
</tbody>
</table>

IF THE LAST COMPONENT OF 'P' IS Omitted, THE NORMAL FIT IS NOT
DOEN, AND THUS THE FIRST AND LAST ROWS AND THE LAST TWO COLUMNS
OF 'T' ARE Omitted.
FR2HOW

TWO-WAY FREQUENCY TABLE
T=P FR2 \n

HISTHOW

FREQUENCY HISTOGRAM
C=M HIST F

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

MYSDHOW

MEAN, VARIANCE, AND STANDARD DEVIATION
T=SYSD X

NTILESHOW

MEDIAN, QUARTILES, ETC.

T*P NTILES F


'NTILES' USES THE SUBFUNCTION 'PBS'.

3. **STP2**

This workspace contains the following functions:

<table>
<thead>
<tr>
<th>NAME</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CM</td>
<td>SIMPLE CORRELATION MATRIX. (CMHOW)</td>
</tr>
<tr>
<td>CORR</td>
<td>SIMPLE AND PARTIAL CORRELATION. (CORRHOW)</td>
</tr>
<tr>
<td>INV</td>
<td>GAUSS-JORDAN MATRIX INVERSION WITH PIVOTING. (INVHOW)</td>
</tr>
<tr>
<td>JINV</td>
<td>GAUSS-JORDAN MATRIX INVERSION. (JINVHOW)</td>
</tr>
<tr>
<td>REG</td>
<td>SIMPLE AND MULTIPLE REGRESSION. (REGHOW)</td>
</tr>
<tr>
<td>RES</td>
<td>CALCULATION OF RESIDUALS. (RESHOW)</td>
</tr>
<tr>
<td>SCORG</td>
<td>CALCULATION OF SIMPLE CORRELATION COEFFICIENTS. (SCORGHOW)</td>
</tr>
<tr>
<td>SR</td>
<td>SIMPLE REGRESSION ANALYSIS. (SRHOW)</td>
</tr>
<tr>
<td>STATRES</td>
<td>RESIDUAL STATISTICS. (STATRESHOW)</td>
</tr>
<tr>
<td>STEG</td>
<td>STEPPED REGRESSION ANALYSIS. (STREGHOW)</td>
</tr>
</tbody>
</table>
CMROW

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.

CORROW

SIMPLE AND PARTIAL CORRELATION
C=V CORR M

'C' IS THE SQUARE MATRIX OF SIMPLE CORRELATION COEFFICIENTS GIVEN BY THE RESULT 'M' OF THE FUNCTION 'CM'. 'V' IS A VECTOR SPECIFYING THE CORRELATION COEFFICIENT TO BE CALCULATED. FOR EXAMPLE, SUPPOSE 'M' IS OF ORDER 6, THEN IF V=(3,6), 'C' IS THE SIMPLE CORRELATION COEFFICIENT BETWEEN VARIABLES 3 AND 6; IF V=(4,1,5,2), 'C' IS THE PARTIAL CORRELATION COEFFICIENT BETWEEN VARIABLES 5 AND 1 WITH THE EFFECTS OF VARIABLES 4 AND 2 REMOVED. 'CORR' USES THE SUBFUNCTION 'INV'.

INROW

GAUSS-JORDAN MATRIX INVERSION WITH PIVOTING
R=LINV R

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

JINROW

GAUSS-JORDAN MATRIX INVERSION
R=JINV R

'RE' IS THE INVERSE OF THE NON-SINGULAR SQUARE MATRIX 'RA' CALCULATED BY THE GAUSS-JORDAN METHOD OF MATRIX INVERSION.
REGRESION

SIMPLE AND MULTIPLE REGRESSION
T< V REG X

'X' IS A MATRIX OF OBSERVATIONS, WHERE THE COLUMNS CORRESPOND TO
VARIABLES 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, 1). THEN 'T'
GIVES THE RESULTS OF THE BEST LEAST-SQUARES FIT OF THE FUNCTION:

X6=A+B*X3+C*X5+D*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: 0, D, ST ERROR OF D, T-VALUE, 0
ROW 5: 0, DF FOR REGRESSION, SUM OF SQUARES, MEAN SQUARE, P-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
ESTIMATES, SQUARE OF MULTIPLE CORRELATION COEFFICIENT

'REG' USES THE SUBFUNCTION 'TVP'.

RESIDUALS
REG REG 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=SCORE 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. 'K' IS A MATRIX WITH 'K' COLUMNS IN THE FOLLOWING
FORM:

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 J AND J

IF 'P' HAS 'K' COLUMNS, THEN 'R' HAS K(K-1)/2 ROWS.

'SCORR' USES THE SUBFUNCTION 'RVS'.

SHOW

SIMPLE REGRESSION
Y=X SR Y

'X' AND 'Y' ARE VECTORS GIVING THE (SAME NUMBER OF) OBSERVATIONS
OF AN INDEPENDENT VARIABLE 'X' AND A DEPENDENT VARIABLE 'Y'. 'Y'
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.
'Y' 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
STATSHOW

RESIDUAL STATISTICS
STATRES T

'T' IS THE MATRIX WITH A 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.

STATSHOW

STANDARD 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 STURVICH ORDER.

THE VECTOR 'Y' 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', 'IVY', AND 'CI'.

4. **ST3**

**THIS WORKSPACE CONTAINS THE FOLLOWING FUNCTIONS:**

<table>
<thead>
<tr>
<th>NAME</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>BINOM</td>
<td>BINOMIAL PROBABILITIES (BINOMHOW)</td>
</tr>
<tr>
<td>CTAB</td>
<td>TWO-WAY CONTINGENCY TABLE (CTABHOW)</td>
</tr>
<tr>
<td>PBS</td>
<td>SUM-SCAN OPERATOR (PBSHOW)</td>
</tr>
<tr>
<td>PERMUTE</td>
<td>PERMUTE VECTOR (PERMUTEROW)</td>
</tr>
<tr>
<td>POISSON</td>
<td>POISSON PROBABILITIES (POISSONHOW)</td>
</tr>
<tr>
<td>RED</td>
<td>ROUNDING (REDHOW)</td>
</tr>
<tr>
<td>SMOOTH</td>
<td>WEIGHTED MOVING AVERAGE (SMOOTHHOW)</td>
</tr>
<tr>
<td>SORT</td>
<td>NUMERIC SORTING (SORTHOW)</td>
</tr>
</tbody>
</table>
BINOMHOW

BINOMIAL PROBABILITIES
B=N BINOM P

CALCULATES THE VECTOR 'P' 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.

CTABNOW

TWO-WAY CONTINGENCY TABLE
R=CTAB T

CALCULATES 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 'RED'.

PRESQ!

SUB-SCAN OPERATOR
V=THE S

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

PERMULHOP

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.
POISSON

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.

RNDNOW

RNDNOW

RND
N+R AND 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 'N', WHICH HAS THE SAME DIMENSIONS AS 'X'.

SMOOTHNOW

WEIGHTED MOVING AVERAGE
V=V SMOOTH X

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

SORTNOW

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:**

<table>
<thead>
<tr>
<th>NAME</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASSIGN</td>
<td>SOLUTION OF MINIMIZATION ASSIGNMENT PROBLEM. (ASSIGNHOW)</td>
</tr>
<tr>
<td>COSTFLOW</td>
<td>SOLUTION OF MINIMUM COST FLOW PROBLEM. (COSTFLOWHOW)</td>
</tr>
<tr>
<td>CPM</td>
<td>CRITICAL PATH ANALYSIS. (CPMHOW)</td>
</tr>
<tr>
<td>CTRANSPORT</td>
<td>SOLUTION OF CAPACITATED TRANSPORTATION PROBLEM. (CTRANSPORTHOW)</td>
</tr>
<tr>
<td>LPSOLN</td>
<td>LINEAR PROGRAMMING SENSITIVITY ANALYSIS. (LPSOLNHOW)</td>
</tr>
<tr>
<td>NETFLOW</td>
<td>FORD-FULKERSON SOLUTION OF A CAPACITATED NETWORK. (NETFLOWHOW)</td>
</tr>
<tr>
<td>RSIM</td>
<td>REVISITED SIMPLEX METHOD LINEAR PROGRAMMING. (RSIMHOW)</td>
</tr>
<tr>
<td>RSIMA</td>
<td>REVISITED SIMPLEX METHOD LINEAR PROGRAMMING WITH CONVERSATIONAL INPUT. (RSIMAHOW)</td>
</tr>
<tr>
<td>SIMPLEX</td>
<td>SIMPLEX METHOD LINEAR PROGRAMMING. (SIMPLEXHOW)</td>
</tr>
<tr>
<td>TRANSPORT</td>
<td>PRIMAL-DUAL SOLUTION OF TRANSPORTATION PROBLEM. (TRANSPORTHOW)</td>
</tr>
</tbody>
</table>
ASSIGNMENT

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.

COSTFLOWHON

MINIMUM COST FLOW
COSTFLOW HON

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'. Each 2×2 matrix 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 'MINFLOW', 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.
CPM HOW

CRITICAL PATH METHOD
CPT-5 CPM NW

'CPM' 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, EASIEST 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 ASTERISK, IN THE MATRIX 'CPT'.


ERROR REPORT: 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.

TRANSPORT HOW

CAPACITATED TRANSPORTATION PROBLEM
S=CAP TRANSPORT COST

THIS FUNCTION USES THE PRIMAL-DUAL ALGORITHM TO SOLVE THE
CAPACITATED TRANSPORTATION PROBLEM. 'COST' IS AN (M+1)X(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;J] IS THE AMOUNT AVAILABLE AT ORIGIN I=1,2,...,M AND
COST[I;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 MxN MATRIX IN WHICH
CAP[I;J] IS THE CAPACITY RESTRICTION ON THE ROUTE FROM ORIGIN 'I'
TO DESTINATION 'J'.

'S' IS AN NxN 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
SUABLE MESSAGE IS GIVEN IN 'S' WHICH IS THEN A MATRIX WITH ONE
ROW, AND 'MINCOST' IS AN EMPTY VECTOR.

'CTRANSPRT' USES THE SUBFUNCTION 'CSTMFLW'.

LP SOL;L;IO;

LINEAR PROGRAMMING SENSITIVITY ANALYSIS
B=LPSONA

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
'SIMPBA' OR 'BSIM', AND THE VECTOR 'B' GIVES THE VARIABLES IN
THE OPTIMAL BASIS. 'LPSONA' 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.

INFEASIBILITY LOWER AND UPPER BOUNDS ARE INDICATED BY VALUES OF
7.237E-05 AND 7.237E-05, RESPECTIVELY.
NETFLOW

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 'E', where 'G' is an N x N capacity matrix in which G[i; j] is the capacity of the directed arc from node 'i' to node 'j'. 'E' 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'.


RSMPH

LINEAR PROGRAMMING
T-197II 

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 7 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 algebraic matrix with a single row.

Each linear programming problem solved with this function must be a maximization problem.
LINEAR PROGRAMMING

RSIN1

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.

'RSIN1' USES THE SUBFUNCTION 'RSIM'.

SIMPLEXNOW

LINEAR PROGRAMMING

RS=B SIMPLEX A


'B' 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 'B' WHICH IS THEN AN ALPHANUMERIC VECTOR.

EACH LINEAR PROGRAMMING PROBLEM SOLVED WITH THIS FUNCTION MUST BE A MAXIMIZING PROBLEM.
TRANSPORTATION PROBLEM
S = TRANSPORT COST

This function uses the primal-dual algorithm to solve the transportation problem. 'COST' is an \((M+1) \times (N+1)\) matrix in which \(\text{COST}[i,j]\) is the unit cost of shipping from origin 'i' to destination 'j', where \(i=1,2,...,M\) and \(j=1,2,...,N\). \(\text{COST}[i,N+1]\) is the amount available at origin 'i', \(i=1,2,...,M\), and \(\text{COST}[M+1;j]\) is the amount required at destination 'j', \(j=1,2,...,N\). \(\text{COST}[M+1;N+1]\) is arbitrary. The sum of the origin availabilities must be equal to the sum of the destination requirements.

'S' is an \(M \times N\) 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. STPS

THIS WORKSPACE CONTAINS FUNCTIONS WHICH PERFORM A CRITICAL PATH ANALYSIS. THE USER ACCESSIONED 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, L.T. 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 FOR EACH NODE, IN THE GLOBAL VARIABLE 'DATA' SO THAT CORRECTIONS CAN 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')
CONVERSATIONAL LINEAR PROGRAMMING
LINP

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 accommodate 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 'RSM' 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 PROBLEM?'.

The program is written in a conversational manner and is designed to be user-friendly.