DOCUMENT RESUME

ED 064 960

24

EM 010 210

TITLE EIN Software Catalog; Entries, Volume III.

INSTITUTION Interuniversity Communications Council (EDUCOM), Boston, Mass.

SPONS AGENCY National Science Foundation, Washington, D.C.; Office of Education (DHEW), Washington, D.C.

BUREAU NO BR-8-0697/Attachment No. 1

PUB DATE [72]

GRANT OEG-0-8-080697-4468(007)

NOTE 563p.; See also EM 010 206, EM 010 207, EM 010 208, EM 010 209, EM 010 211, EM 010 212

EDRS PRICE MF-$0.65 HC-$19.74

DESCRIPTORS Behavioral Sciences; Biological Sciences; *Catalogs; Computer Oriented Programs; *Computer Programs; Earth Science; Educational Administration; Engineering; *Information Networks; Information Science; *Interinstitutional Cooperation; Library Science; Mathematics; Operations Research; Physical Sciences; *Program Descriptions; Statistics; Technology; Tests

IDENTIFIERS *Educational Information Network; EIN

ABSTRACT The EIN (Educational Information Network) is a non-profit operation which coordinates the sharing of educational computing resources. It is administered by EDUCOM and funded jointly by the U.S. Office of Education and the National Science Foundation. EIN maintains a group of contact personnel at member institutions to serve as a liaison between the institution and EIN. Through these persons items of software are offered for distribution. EIN also publishes a catalog of the software which is available through the network. The four-volume catalog contains an alphabetical listing of the participating EIN members that are represented in the catalog by program descriptions; descriptions of each computer facility listed and its general pricing algorithm; abstracts of available programs, subdivided into 13 areas of application; three indexes—by EIN number, by descriptive title, and by keyword; and complete descriptions of programs, including user instructions, samples of input and output, and cost estimates. Volume one contains a description of the EIN, facilities descriptions, abstracts, and indexes. This volume is one of three which contain the complete program descriptions. (JY)
EIN Software Catalog: Entries, Volume III.
FUNCTIONAL ABSTRACT

INOUT will solve the static Leontief input–output problem for a model with as many as 175 sectors. From data consisting of a square input matrix, an output vector, and a set of final demand vectors, a great variety of results may be selectively opted, calculated, and printed. Among these are matrices of technical and interdependency coefficients, matrices of interdependency values and net effects, and some other useful vectors.

REFERENCE

USER INSTRUCTIONS

The data deck consists of the following cards in the order specified.

```
// EXEC LIBRARY.PARM=INOUT
// DATA.INPUT DD *
Master control card for Problem 1
Total output or receipts vector
Input matrix cards
Final demand vector
Master control card for Problem 2
```

Master Control Card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-3</td>
<td>number of sectors (must not exceed 175)</td>
</tr>
<tr>
<td>10</td>
<td>1 if technical coefficients are to be printed (else =0)</td>
</tr>
<tr>
<td>15</td>
<td>1 if interdependency coefficients are to be printed (else = 0)</td>
</tr>
<tr>
<td>16-20</td>
<td>number of final demand vectors to be read</td>
</tr>
<tr>
<td>25</td>
<td>1 if sector multipliers are to be printed (else = 0)</td>
</tr>
<tr>
<td>30</td>
<td>1 if matrix of interdependency values is to be printed (else = 0)</td>
</tr>
<tr>
<td>35</td>
<td>1 if matrix of net effects is to be printed (else = 0)</td>
</tr>
<tr>
<td>40</td>
<td>1 if column sums of technical coefficients are to be printed (else = 0)</td>
</tr>
<tr>
<td>45</td>
<td>1 if interdependency coefficients are to be punched (else = 0)</td>
</tr>
<tr>
<td>50</td>
<td>1 if matrix of technical coefficients is to be punched in 8F10.5 format (else = 0)</td>
</tr>
<tr>
<td>55</td>
<td>1 if matrix of interdependency coefficients is to be punched in 4F20.8 format (else = 0)</td>
</tr>
</tbody>
</table>

Total Output or Receipts Vector Cards

This vector is punched eight elements per card in 10-column fields. If the decimal point is not punched, it will automatically be continued.
placed between the sixth and seventh columns in each 10-column field. These cards are to follow the master control card.

**Input-Matrix Cards**

The input matrix is punched by rows, eight elements per card in 10-column fields. The first element in each row will occupy the first 10-column field on a new card—each subsequent element in that row will follow in consecutive order. If the decimal point is not punched, it will automatically be placed between the sixth and seventh columns in each 10-column field. These cards are to follow the total output vector cards.

**Final Demand Vector(s) Cards**

If a nonzero number appears in Cols. 16-20 of the master control card, the final demand vectors desired will be punched in the same fashion as the total output or receipts vector. Each demand vector will require a separate set of cards. These cards will follow the input-matrix cards. To obtain solutions for more than one final demand vector, insert a set of cards for each final demand vector and make the appropriate entry in Cols. 16-20 of the master control card.
EDUCOM

SAMPLE INPUT

// PROCEDURE

// PARAMETERS

// MAIN

// DATA

// OUTPUT

// END
## Sample Output

### Sector Input-Output Model

#### Matrix of Technical Coefficients

$$A = \begin{bmatrix} 0.3424022 & 0.1619718 & 0.0 & 0.00745941 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0059107 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.5 & 0.0056049 & 0.0 & 0.0 & 0.0 \\ \end{bmatrix}$$

#### Inverse Matrix of Interdependence Coefficients

$$A^{-1} = \begin{bmatrix} 0.0324022 & 0.1661971 & 0.0 & 0.02788104 & 0.0 & 0.0 & 0.00266312 \\ 0.0 & 0.0056049 & 0.0 & 0.01292673 & 0.0 & 0.0 & 0.00445693 \\ 0.0 & 0.00468591 & 0.0 & 0.00059102 & 0.0 & 0.0 & 0.00080800 \\ 0.0 & 0.0056049 & 0.0 & 0.01292673 & 0.0 & 0.0 & 0.00445693 \\ \end{bmatrix}$$

### Technical Coefficients

<table>
<thead>
<tr>
<th>Sector</th>
<th>A Matrix of Technical Coefficients</th>
<th>Inverse Matrix of Interdependence Coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sector 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sector 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sector 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sector 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sector 5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**Continued...**
<table>
<thead>
<tr>
<th>SECTOR</th>
<th>COLUMN SUMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>C.40064461</td>
</tr>
<tr>
<td>2</td>
<td>C.06015557</td>
</tr>
<tr>
<td>3</td>
<td>C.04170378</td>
</tr>
<tr>
<td>4</td>
<td>C.06475554</td>
</tr>
<tr>
<td>5</td>
<td>C.06716575</td>
</tr>
<tr>
<td>6</td>
<td>C.06514556</td>
</tr>
<tr>
<td>7</td>
<td>C.06575632</td>
</tr>
<tr>
<td>8</td>
<td>C.06591567</td>
</tr>
<tr>
<td>9</td>
<td>C.06875834</td>
</tr>
<tr>
<td>10</td>
<td>C.04524569</td>
</tr>
<tr>
<td>11</td>
<td>C.06645672</td>
</tr>
<tr>
<td>12</td>
<td>C.06725745</td>
</tr>
<tr>
<td>13</td>
<td>C.06725745</td>
</tr>
<tr>
<td>14</td>
<td>C.06725745</td>
</tr>
<tr>
<td>15</td>
<td>C.06725745</td>
</tr>
<tr>
<td>16</td>
<td>C.06725745</td>
</tr>
<tr>
<td>17</td>
<td>C.06725745</td>
</tr>
<tr>
<td>18</td>
<td>C.06725745</td>
</tr>
<tr>
<td>19</td>
<td>C.06725745</td>
</tr>
<tr>
<td>20</td>
<td>C.06725745</td>
</tr>
<tr>
<td>21</td>
<td>C.06725745</td>
</tr>
<tr>
<td>22</td>
<td>C.06725745</td>
</tr>
<tr>
<td>23</td>
<td>C.06725745</td>
</tr>
<tr>
<td>24</td>
<td>C.06725745</td>
</tr>
<tr>
<td>25</td>
<td>C.06725745</td>
</tr>
<tr>
<td>26</td>
<td>C.06725745</td>
</tr>
<tr>
<td>27</td>
<td>C.06725745</td>
</tr>
<tr>
<td>28</td>
<td>C.06725745</td>
</tr>
<tr>
<td>29</td>
<td>C.06725745</td>
</tr>
</tbody>
</table>
## Educational Information Network

### Interdependence Coefficients

#### Sector 1

<table>
<thead>
<tr>
<th>Sector 1</th>
<th>(x_1)</th>
<th>(y_1)</th>
<th>(z_1)</th>
<th>(w_1)</th>
<th>(u_1)</th>
<th>(v_1)</th>
<th>(t_1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.03438313</td>
<td>0.07183136</td>
<td>0.00231460</td>
<td>0.00343522</td>
<td>(x_2)</td>
<td>(y_2)</td>
<td>(z_2)</td>
<td>(w_2)</td>
</tr>
</tbody>
</table>

#### Sector 2

<table>
<thead>
<tr>
<th>Sector 2</th>
<th>(x_3)</th>
<th>(y_3)</th>
<th>(z_3)</th>
<th>(w_3)</th>
<th>(u_3)</th>
<th>(v_3)</th>
<th>(t_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00190291</td>
<td>0.00209732</td>
<td>0.00462414</td>
<td>(x_4)</td>
<td>(y_4)</td>
<td>(z_4)</td>
<td>(w_4)</td>
<td>(u_4)</td>
</tr>
</tbody>
</table>

#### Sector 3

<table>
<thead>
<tr>
<th>Sector 3</th>
<th>(x_5)</th>
<th>(y_5)</th>
<th>(z_5)</th>
<th>(w_5)</th>
<th>(u_5)</th>
<th>(v_5)</th>
<th>(t_5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_6)</td>
<td>(y_6)</td>
<td>(z_6)</td>
<td>(w_6)</td>
<td>(u_6)</td>
<td>(v_6)</td>
<td>(t_6)</td>
<td></td>
</tr>
</tbody>
</table>

#### Sector 4

<table>
<thead>
<tr>
<th>Sector 4</th>
<th>(x_7)</th>
<th>(y_7)</th>
<th>(z_7)</th>
<th>(w_7)</th>
<th>(u_7)</th>
<th>(v_7)</th>
<th>(t_7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_8)</td>
<td>(y_8)</td>
<td>(z_8)</td>
<td>(w_8)</td>
<td>(u_8)</td>
<td>(v_8)</td>
<td>(t_8)</td>
<td></td>
</tr>
</tbody>
</table>

### Note

- Interdependence coefficients indicate the degree to which one sector depends on another.
- Each coefficient represents the flow of information or data from one sector to another.
- Values are shown as decimal numbers, indicating the strength of the interdependence.

---

**Educational Information Network**

**EDUCOM**

---

**continued**
<table>
<thead>
<tr>
<th>Sector</th>
<th>Multiplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.78665121</td>
</tr>
<tr>
<td>2</td>
<td>2.61755447</td>
</tr>
<tr>
<td>3</td>
<td>3.57165621</td>
</tr>
<tr>
<td>4</td>
<td>4.43296646</td>
</tr>
<tr>
<td>5</td>
<td>5.29363651</td>
</tr>
<tr>
<td>6</td>
<td>6.18646849</td>
</tr>
<tr>
<td>7</td>
<td>7.16455274</td>
</tr>
<tr>
<td>8</td>
<td>8.13546590</td>
</tr>
<tr>
<td>9</td>
<td>9.12856871</td>
</tr>
<tr>
<td>10</td>
<td>10.11557917</td>
</tr>
<tr>
<td>11</td>
<td>11.10465565</td>
</tr>
<tr>
<td>12</td>
<td>12.99777777</td>
</tr>
<tr>
<td>13</td>
<td>13.98888888</td>
</tr>
<tr>
<td>14</td>
<td>14.98888888</td>
</tr>
<tr>
<td>15</td>
<td>15.13144321</td>
</tr>
<tr>
<td>16</td>
<td>16.77555555</td>
</tr>
<tr>
<td>17</td>
<td>17.70537481</td>
</tr>
<tr>
<td>18</td>
<td>18.16040871</td>
</tr>
<tr>
<td>19</td>
<td>19.92123123</td>
</tr>
<tr>
<td>20</td>
<td>20.14053911</td>
</tr>
<tr>
<td>21</td>
<td>21.92154154</td>
</tr>
<tr>
<td>22</td>
<td>22.51232123</td>
</tr>
<tr>
<td>23</td>
<td>23.12345567</td>
</tr>
<tr>
<td>24</td>
<td>24.78954444</td>
</tr>
<tr>
<td>25</td>
<td>25.78954444</td>
</tr>
<tr>
<td>26</td>
<td>26.78954444</td>
</tr>
<tr>
<td>27</td>
<td>27.17985333</td>
</tr>
<tr>
<td>28</td>
<td>28.16535353</td>
</tr>
<tr>
<td>29</td>
<td>29.78954444</td>
</tr>
<tr>
<td>30</td>
<td>30.17985333</td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th>SECTOR</th>
<th>FINAL DEMAND</th>
<th>REQUIRED OUTPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>144,000,000</td>
<td>179,000,000</td>
</tr>
<tr>
<td>2</td>
<td>99,000,000</td>
<td>95,000,000</td>
</tr>
<tr>
<td>3</td>
<td>29,000,000</td>
<td>23,000,000</td>
</tr>
<tr>
<td>4</td>
<td>156,000,000</td>
<td>156,000,000</td>
</tr>
<tr>
<td>5</td>
<td>111,000,000</td>
<td>89,000,000</td>
</tr>
<tr>
<td>6</td>
<td>23,000,000</td>
<td>778,000,000</td>
</tr>
<tr>
<td>7</td>
<td>585,000,000</td>
<td>167,000,000</td>
</tr>
<tr>
<td>8</td>
<td>345,000,000</td>
<td>167,000,000</td>
</tr>
<tr>
<td>9</td>
<td>71,000,000</td>
<td>201,000,000</td>
</tr>
<tr>
<td>10</td>
<td>326,000,000</td>
<td>149,000,000</td>
</tr>
<tr>
<td>11</td>
<td>716,000,000</td>
<td>378,000,000</td>
</tr>
<tr>
<td>12</td>
<td>108,000,000</td>
<td>96,000,000</td>
</tr>
<tr>
<td>13</td>
<td>111,000,000</td>
<td>111,000,000</td>
</tr>
<tr>
<td>14</td>
<td>165,000,000</td>
<td>121,000,000</td>
</tr>
<tr>
<td>15</td>
<td>219,000,000</td>
<td>219,000,000</td>
</tr>
<tr>
<td>16</td>
<td>32,000,000</td>
<td>32,000,000</td>
</tr>
<tr>
<td>17</td>
<td>194,000,000</td>
<td>194,000,000</td>
</tr>
<tr>
<td>18</td>
<td>111,000,000</td>
<td>111,000,000</td>
</tr>
<tr>
<td>19</td>
<td>71,000,000</td>
<td>71,000,000</td>
</tr>
<tr>
<td>20</td>
<td>119,000,000</td>
<td>119,000,000</td>
</tr>
<tr>
<td>21</td>
<td>617,000,000</td>
<td>829,000,000</td>
</tr>
<tr>
<td>22</td>
<td>32,000,000</td>
<td>32,000,000</td>
</tr>
<tr>
<td>23</td>
<td>122,000,000</td>
<td>149,000,000</td>
</tr>
<tr>
<td>24</td>
<td>294,000,000</td>
<td>497,000,000</td>
</tr>
<tr>
<td>25</td>
<td>44,000,000</td>
<td>104,000,000</td>
</tr>
<tr>
<td>26</td>
<td>215,000,000</td>
<td>554,000,000</td>
</tr>
<tr>
<td>27</td>
<td>16,000,000</td>
<td>311,000,000</td>
</tr>
<tr>
<td>28</td>
<td>104,000,000</td>
<td>117,000,000</td>
</tr>
<tr>
<td>29</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>TOTAL</td>
<td>1,238,000,000</td>
<td>2,544,000,000</td>
</tr>
</tbody>
</table>
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 11 seconds. At the current rate for computer time ($0.11/sec.) at The Pennsylvania State University, the chargeable time amounts to $1.21.

Charge to user = computer time + network overhead
               = $1.21 + network overhead

CONTENTS—INOUT

pages  Identification & Abstract
       1
       3- 4  User Instructions
       5-12  I/O
       13  Cost—Contents
The Synagraphic Mapping program (SYMAP) produces maps that depict spatially disposed quantitative and qualitative information. Raw data of every kind (physical, social, economic, etc.) may be related, weighted, and aggregated in a graphic format by assigning values to the coordinate locations of data points or data zones. According to the application and desired representation of data, three basic types of mapping procedure may be specified: contour, conformant, or proximal.

**CONTOUR**
—based on the use of *contour* lines, each of which represents a value remaining constant throughout its length. The map consists of closed curves that connect all points having the same numerical value. The value at each of the different levels of contour...
(where a single contour level will represent a particular data value) is determined by the program, according to the scale of the map and the range of the data. Between any two contour lines, a continuous variation is assumed. Therefore, the use of contour mapping should be restricted to the representation of continuously varying information, such as topography, rainfall, or population density.

**CONFORMANT**
—based on the conformance to the boundaries of a data zone. This type of mapping is best suited to data for which the representation as a continuously varying surface is inappropriate owing to the significance of areal limits or boundaries. Each predefined data zone is assigned one data value and, depending on its numeric class (range), one representative character on the map itself. Local variation within the zone boundaries will not be apparent, but will, on the average, be correct.

**PROXIMAL**
—based on proximity to a data point. In appearance, this type of map is similar to the conformant map. However, point information is used here to define the data zones. Each character location on the output map is assigned the value of the nearest data point, using nearest-neighbor techniques. Boundaries are then assumed along the lines where these values change. Then, the mapping is carried out as in the conformant type.

While the contour type of map is most often used and the easiest to produce, the conformant and proximal maps are often more helpful in the "soft" disciplines. Output is in the form of printed pages that, if the total map size exceeds the width of the computer printed page (13 in.), may easily be glued or pasted together to form a continuous map. Also included in the output is a histogram showing frequencies for given data levels, plus several optional features.

**REFERENCES**


continued
Shepard, D., "A Two-Dimensional Interpolation Function for Irregularly Spaced Data," Harvard Univ. Grad. School Design Lab. Computer Graphics (Feb. 1968); available from the EIN Office for the cost of duplication and mailing. Deals with the subject of analyzing irregularly spaced data derived from a continuous surface. A method is developed for reconstructing the surface from the sampled data. This method is the main device used to generate the maps produced by SYMAP.
USER INSTRUCTIONS

The user must establish the geographic area that he wishes to map, called the study area; it may be of any desired shape. He must then establish the relative locations of any controlling points to which his data relate. For these two purposes, it is best to construct a source map, with borders from which measurements may be taken. This source map may be identical in size to the maps that are to be produced—or it may be of any size, because there are automatic scaling and proportioning electives built into the program.

Owing to the availability of the digitizer at The Pennsylvania State University (PSU), it is recommended that users of the PSU modification of SYMAP consider using this device for these proportional adjustments. The purpose of the digitizer is to transfer automatically to punched cards, the X and Y coordinates from a map or drawing. The drawing or map itself is limited to 30X30 in., but it may be drawn to almost any scale. The digitizer is in no way connected to a computer but only provides the cards (via an IBM 029 card-punch machine) that can be used as computer input. Interested persons are directed to contact the EIN technical representative at PSU.

Input to the computer in the form of a deck of punched cards must be prepared. This deck will consist of certain introductory cards and a number of packages, each composed of additional cards covering a specific category of information about the map to be produced.

Job Card
Will be prepared by Computation Center personnel.

Introductory Cards
See Ref. 1.

Types of Packages
The titles of all available packages, with a brief explanation of their general purpose, are listed below in the sequence of their position in the deck. For more-complete and definitive instructions in the preparation of these packages, see Ref. 1.
A-OUTLINE
—describes the outline of the study area if nonrectangular by specifying the coordinate locations of the outline vertices. Used for contour and proximal maps only.

A-CONFORMOLINES
—used to give the positions of the data zones to which your data are to be related, by specifying the coordinate locations of vertices on the zonal outlines. This package is required for a conformant map.

B-DATA POINTS
—used to give the positions of the data points to which your data are to be related, by specifying their coordinate locations. Data points may be either the points for which data are available or the centers of areas, called data zones, for which data are available. (When warranted by the nature of your study, and under exceptional circumstances, other "centers" may be used, such as centers of population.) This package is required for contour and proximal maps.

C-OTOLEGENDS
—used to specify the relative position of legends that are to be adjusted automatically if the size and/or scale of the map are altered.

D-BARRIERS
—used to give the coordinate location and strength of impediments to interpolation at specified vertices.

E-VALUES
—used to assign numerical data to the data points and/or data zones by specifying the values involved. All such data must, of course, be measured on a consistent uniform basis. While normally required, this package may be omitted if you wish to procure a preliminary base map for checking locations before applying values.

E1-VALUES INDEX
—used to adjust the reference order of data values in the E-VALUES package.

F-MAP
—used to specify below the map an appropriate title for the identification of each separate map that you wish to run. In addition, it instructs the computer to make each specific map pursuant to
certain electives. These electives provide a variety of options for obtaining maps suited to your particular needs. An F-MAP package is required for each map desired.

REFERENCES

SAMPLE INPUT—Population-Density Contour Map of Pennsylvania

'C993,T=200,R=600,SYMAP','HULL E J',MSGLEVEL=1
// EXEC LIBRARY,PARM=SYMAP
//DATA.INPUT DD n
A-OUTLINES
0008 0052
0027 0051
0027 0059
0028 0093
0028 0131
0028 0161
0028 0197
0028 0237
0028 0278
0028 0304
0087 0301

B-DATA
0018 0036
0039 0082
0042 0119
0044 0140
0046 0177
0086 0085
0096 0119

C-OTOLEGENS
PHILADELPHIA
10 P 0135 0034 -5. -4.
PITTSBURGH
15 P 0111 0140 10

PENN STATE
TL 0099 0008
0126 0046
0138 0053
0147 0054
0157 0069
0152 0295
0155 0316
IL 0082 0009
0081 0043
0084 0061
0082 0068
0084 0083
0096 0283
0096 0293
0087 0141
0078 0148
0074 0159
0088 0167
0081 0186
0176 0227
0190 0238
99999
continued
### E-VALUES

<table>
<thead>
<tr>
<th>E-VALUES</th>
</tr>
</thead>
<tbody>
<tr>
<td>309.0</td>
</tr>
<tr>
<td>50.0</td>
</tr>
<tr>
<td>55.0</td>
</tr>
<tr>
<td>15.0</td>
</tr>
<tr>
<td>52.0</td>
</tr>
<tr>
<td>46.0</td>
</tr>
<tr>
<td>40.0</td>
</tr>
<tr>
<td>35.0</td>
</tr>
<tr>
<td>25.0</td>
</tr>
<tr>
<td>21.0</td>
</tr>
<tr>
<td>20.0</td>
</tr>
<tr>
<td>42.0</td>
</tr>
<tr>
<td>15.0</td>
</tr>
<tr>
<td>40.0</td>
</tr>
<tr>
<td>9.0</td>
</tr>
<tr>
<td>42.0</td>
</tr>
<tr>
<td>9.0</td>
</tr>
<tr>
<td>46.0</td>
</tr>
<tr>
<td>11.0</td>
</tr>
<tr>
<td>117.0</td>
</tr>
<tr>
<td>77.0</td>
</tr>
<tr>
<td>187.0</td>
</tr>
<tr>
<td>108.0</td>
</tr>
<tr>
<td>144.0</td>
</tr>
<tr>
<td>63.0</td>
</tr>
<tr>
<td>72.0</td>
</tr>
<tr>
<td>71.0</td>
</tr>
</tbody>
</table>

### F-MAP

#### PENNSYLVANIA POPULATION DENSITY

<table>
<thead>
<tr>
<th>C Persons per Square Mile</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>24</td>
</tr>
<tr>
<td>36</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>25</td>
</tr>
<tr>
<td>25</td>
</tr>
<tr>
<td>100</td>
</tr>
<tr>
<td>100</td>
</tr>
<tr>
<td>100</td>
</tr>
<tr>
<td>100</td>
</tr>
<tr>
<td>100</td>
</tr>
<tr>
<td>100</td>
</tr>
<tr>
<td>100</td>
</tr>
<tr>
<td>100</td>
</tr>
<tr>
<td>14584</td>
</tr>
<tr>
<td>14</td>
</tr>
<tr>
<td>1.1</td>
</tr>
<tr>
<td>1.5</td>
</tr>
<tr>
<td>2.0</td>
</tr>
<tr>
<td>1.5</td>
</tr>
</tbody>
</table>

1965 DATA

C. F. GERLACH

9999

13

14

99999

1/70
SAMPLE INPUT—Population-Density Conformant Map of Pennsylvania

```sql
// 'C5993,T=200.,R=6000.,SYMAP', 'HULL E J', MSGLEVEL=1
// EXEC LIBRARY,PARMS=SYMAP
// DATA, INPUT DD *
VERSION 5
A-CONFORMOLINES
A  0008  0052 01
  0027  0059 02
  0037  0059 03
  0046  0012 04
  0027  0017 05
  0023  0023 06
  0008  0052 07

A  0037  0059 08
  0040  0095 09
  0040  0093 10
  0054  0093 11
99999
B-DATA
  0018  0036 01
  0019  0082 02
  0069  0019 03
  0081  0019 04
99999
C-LEGENDS
  12 P  0165  0297
PHILADELPHIA  10 P  0135  0034 -5.
  15 P  0111  0149
PITTSBURGH  PENN STATE
  0099  0008 10
  0343  0199 01
  0347  0198 02
  0176  0227 03
  0190  0238 04
```

continued
E-VALUES

300.0
50.0
55.0
15.0
32.0
48.0
40.0
38.0
17.0
517.0
42.0
15.0
90.0
42.0
15.0
32.0
46.0
11.0
17.0
77.0
187.0
300.0
144.0
63.0
72.0
71.0

F-MAP

C PENNSYLVANIA POPULATION DENSITY
C PERSONS PER SQUARE MILE

1 24.
3 10.
4 25.
5 1000.
6 25. 25. 50. 100. 100. 100.
10 100. 100. 400. 2000.

1985 DATA
C. F. GERLACH

9999
13 .11
14 1.5 1.5 2.0 1.5
15 6.10.

99999

/\
SAMPLE OUTPUT—Showing Contours of Population Density

```
VERTEX    DOWN    ACROSS
         1
(1)  27.00  11.00
(2)  4.70  52.00
(3)  23.00  50.00
(4)  24.30  55.00
(5)  29.30  131.00
(6)  28.30  130.00
(7)  29.00  17.00
(8)  29.50  277.00
(9)  28.50  776.00

```

```
POINT    DOWN    ACROSS
        1
(1)  19.00  35.00
(2)  39.00  62.00
(3)  42.00  119.00
(4)  44.00  140.00
(5)  46.00  177.00
(6)  45.00  312.00
(7)  40.00  291.00
(8)  67.00  395.00
(9)  73.00  406.00
(10) 70.00  273.00
```

```
(60) 144.00  231.00
(61) 143.00  243.00
(62) 134.00  271.00
(63) 144.00  202.00
(64) 157.00  281.00
(65) 148.00  268.00
(66) 171.00  270.00
(67) 165.00  271.00
```

continued
<table>
<thead>
<tr>
<th>VERTEX</th>
<th>DOWN</th>
<th>ACROSS</th>
<th>ROWS</th>
<th>COLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>PHILADELPHIA</td>
<td>ACROSS FROM</td>
<td>166.00</td>
<td>297.00</td>
</tr>
<tr>
<td>2)</td>
<td>PITTSBURGH</td>
<td>DOWN FROM</td>
<td>135.00</td>
<td>34.00</td>
</tr>
<tr>
<td>3)</td>
<td>PENN STATE</td>
<td>ACROSS FROM</td>
<td>111.00</td>
<td>149.00</td>
</tr>
</tbody>
</table>

(4) **ON LINE**

| 1) | 99.00 | 8.00 |
| 2) | 126.00 | 46.00 |
| 3) | 136.00 | 51.00 |
| 4) | 147.00 | 54.00 |
| 5) | 157.00 | 69.00 |

(6) **ON LINE**

| 1) | 82.00 | 9.00 |
| 2) | 81.00 | 41.00 |
| 3) | 84.00 | 63.00 |
| 4) | 82.30 | 68.00 |
| 5) | 44.00 | 83.00 |

(16) 103.00 154.00
(17) 148.30 218.00
(18) 161.00 247.00
(19) 132.00 295.00
(20) 155.00 316.00

LENGTH: 351.82

(5) **ON LINE**

| 1) | 82.00 | 9.00 |
| 2) | 81.00 | 41.00 |
| 3) | 84.00 | 63.00 |
| 4) | 82.30 | 68.00 |
| 5) | 44.00 | 83.00 |

(16) 98.00 209.00
(17) 94.00 228.00
(18) 97.00 235.00
(19) 93.00 266.00
(20) 92.00 280.00
(21) 96.00 295.00
(22) 96.00 295.00

LENGTH: 297.00

(6) **ON LINE**

| 1) | 87.00 | 144.00 |
| 2) | 78.00 | 144.00 |
| 3) | 74.00 | 155.00 |
| 4) | 80.00 | 167.00 |
| 5) | 81.00 | 186.00 |

continued
### F-VALUES

<table>
<thead>
<tr>
<th>DATUM</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>109.00</td>
</tr>
<tr>
<td>2</td>
<td>40.00</td>
</tr>
<tr>
<td>3</td>
<td>55.00</td>
</tr>
<tr>
<td>4</td>
<td>15.20</td>
</tr>
<tr>
<td>5</td>
<td>12.00</td>
</tr>
<tr>
<td>6</td>
<td>46.00</td>
</tr>
<tr>
<td>7</td>
<td>40.00</td>
</tr>
<tr>
<td>8</td>
<td>38.00</td>
</tr>
<tr>
<td>9</td>
<td>17.00</td>
</tr>
<tr>
<td>10</td>
<td>417.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DATUM</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>161.00</td>
</tr>
<tr>
<td>202</td>
<td>163.00</td>
</tr>
<tr>
<td>203</td>
<td>173.00</td>
</tr>
<tr>
<td>204</td>
<td>176.00</td>
</tr>
<tr>
<td>205</td>
<td>140.00</td>
</tr>
<tr>
<td>LENGTH</td>
<td>219.26</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DATUM</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>601</td>
<td>244.00</td>
</tr>
<tr>
<td>611</td>
<td>330.30</td>
</tr>
<tr>
<td>621</td>
<td>654.30</td>
</tr>
<tr>
<td>631</td>
<td>503.00</td>
</tr>
<tr>
<td>641</td>
<td>1042.00</td>
</tr>
<tr>
<td>651</td>
<td>277.00</td>
</tr>
<tr>
<td>661</td>
<td>200.00</td>
</tr>
<tr>
<td>671</td>
<td>15594.00</td>
</tr>
</tbody>
</table>

*continued*
EDUCATIONAL INFORMATION NETWORK

000 0047

C-Pennsylvania Population Density
C-Persons per square mile

SELECTIVF

MAP SIZE IS 24.00 INCHES LONG BY 36.00 INCHES WIDE
3 NUMBER OF LEVELS IS 10
6 LEVEL SIZES ARE PROPORTIONAL TO 4) 25.00 7) 100.00
5) 50.00 8) 100.00 6) 100.00 9) 100.00 10) 1458.00
10 MAP TEXT
C.F. GEHLACH
13 MAP SCALE IS 0.11
14 MAP SHIFTS ARE 1.00 AT TOP, 1.50 AT LEFT, 2.00 AT BOTTOM, AND 1.50 AT RIGHT
15 NEW NUMBER OF LINES PER INCH IS 4.00

TIME = 19:59:13.12
ELAPSE TIME = 3.37 SECONDS FOR INPUT

continued
MAP 1

PENNSYLVANIA POPULATION DENSITY
PERSONS PER SQUARE MILE

MAP SCALE = 0.1100 INCHES ON OUTPUT MAP/UNITS ON SOURCE MAP
MAP SHOULD BE PRINTED AT 5.0 ROWS PER INCH AND 10.0 COLUMNS PER INCH

ROW = (DOWN COORDINATE - 0.82) * 0.4400
COLUMN = (ACROSS COORDINATE - 1.44) * 1.1000

DATA POINTS FOR MAP

<table>
<thead>
<tr>
<th>POINT</th>
<th>ROW</th>
<th>COLUMN</th>
<th>DATUM</th>
<th>VALUE</th>
<th>LEVEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14</td>
<td>43</td>
<td>1</td>
<td>309.00</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>32</td>
<td>74</td>
<td>2</td>
<td>50.00</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>34</td>
<td>174</td>
<td>3</td>
<td>55.00</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>36</td>
<td>157</td>
<td>4</td>
<td>15.00</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>37</td>
<td>198</td>
<td>5</td>
<td>32.00</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>34</td>
<td>240</td>
<td>6</td>
<td>44.00</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>33</td>
<td>240</td>
<td>7</td>
<td>40.00</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>44</td>
<td>315</td>
<td>8</td>
<td>14.00</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>55</td>
<td>360</td>
<td>9</td>
<td>17.00</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>57</td>
<td>291</td>
<td>10</td>
<td>417.00</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>101</td>
<td>278</td>
<td>61</td>
<td>320.00</td>
<td>6</td>
</tr>
<tr>
<td>62</td>
<td>93</td>
<td>304</td>
<td>62</td>
<td>654.00</td>
<td>9</td>
</tr>
<tr>
<td>63</td>
<td>107</td>
<td>324</td>
<td>63</td>
<td>530.00</td>
<td>7</td>
</tr>
<tr>
<td>64</td>
<td>110</td>
<td>313</td>
<td>64</td>
<td>1042.00</td>
<td>10</td>
</tr>
<tr>
<td>65</td>
<td>117</td>
<td>290</td>
<td>65</td>
<td>277.00</td>
<td>5</td>
</tr>
<tr>
<td>66</td>
<td>119</td>
<td>310</td>
<td>66</td>
<td>300.00</td>
<td>10</td>
</tr>
<tr>
<td>67</td>
<td>118</td>
<td>394</td>
<td>67</td>
<td>15584.00</td>
<td>10</td>
</tr>
</tbody>
</table>

STANDARD SEARCH RADIUS IS 42.9321
TIME = 18:13:13.59
ELAPSE TIME = 0.47 SECONDS FOR INITIAL CALCULATIONS

continued
TIME = 19:15:52.71
ELAPSED TIME = 167.12 SECONDS FOR MAP

C PENNSYLVANIA POPULATION DENSITY
C PERSONS PER SQUARE MILE

DATA VALUE EXTREMES ARE 11.00 15584.00

ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL
(MAXIMUM INCLUDED IN HIGHEST LEVEL ONLY)

| MINIMUM | 11.00 | 15.96 | 110.93 | 210.96 | 310.79 | 410.72 | 510.65 | 615.68 | 1010.20 |
| MAXIMUM | 35.93 | 60.96 | 110.93 | 210.86 | 313.70 | 410.72 | 510.65 | 615.68 | 15584.00 |

PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL

<table>
<thead>
<tr>
<th>LEVEL</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.16</td>
<td>0.16</td>
<td>0.32</td>
<td>0.64</td>
<td>0.64</td>
<td>0.64</td>
<td>0.64</td>
<td>2.57</td>
<td>93.58</td>
<td></td>
</tr>
</tbody>
</table>

FREQUENCY DISTRIBUTION OF DATA POINT VALUES IN EACH LEVEL

<table>
<thead>
<tr>
<th>LEVEL</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>FREQ.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TIME = 18:19:52.66
ELAPSED TIME = 1.09 SECONDS FOR HISTOGRAM
### SAMPLE OUTPUT—Showing Population Density by Counties

#### A-COMPLETE MATICS

<table>
<thead>
<tr>
<th>VERT</th>
<th>DOWN</th>
<th>ACROSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>92.00</td>
<td></td>
</tr>
<tr>
<td>2.00</td>
<td>91.00</td>
<td></td>
</tr>
<tr>
<td>3.00</td>
<td>90.00</td>
<td></td>
</tr>
<tr>
<td>4.00</td>
<td>59.00</td>
<td></td>
</tr>
<tr>
<td>5.00</td>
<td>12.00</td>
<td></td>
</tr>
<tr>
<td>6.00</td>
<td>12.00</td>
<td></td>
</tr>
<tr>
<td>7.00</td>
<td>23.00</td>
<td></td>
</tr>
<tr>
<td>8.00</td>
<td>42.00</td>
<td></td>
</tr>
<tr>
<td>9.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### AREA

- Length: 137.99
- Area: 792.50
- Center: 264.9N, 37.26E

#### AREA

- Length: 44.89
- Area: 137.09
- Center: 169.6N, 791.93

#### DATA POINTS

<table>
<thead>
<tr>
<th>PRINT</th>
<th>DOWN</th>
<th>ACROSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8.00</td>
<td>92.00</td>
</tr>
<tr>
<td>2</td>
<td>38.00</td>
<td>39.00</td>
</tr>
<tr>
<td>3</td>
<td>42.00</td>
<td>115.00</td>
</tr>
<tr>
<td>4</td>
<td>44.00</td>
<td>140.00</td>
</tr>
<tr>
<td>5</td>
<td>46.00</td>
<td>177.00</td>
</tr>
<tr>
<td>6</td>
<td>52.00</td>
<td>252.00</td>
</tr>
<tr>
<td>7</td>
<td>40.00</td>
<td>257.00</td>
</tr>
<tr>
<td>8</td>
<td>47.00</td>
<td>294.00</td>
</tr>
<tr>
<td>9</td>
<td>70.00</td>
<td>363.00</td>
</tr>
<tr>
<td>10</td>
<td>70.00</td>
<td>263.00</td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th>VERTEX</th>
<th>PUMP</th>
<th>MEET</th>
<th>CONVEY</th>
<th>EDGES</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 1 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 2 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 3 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 4 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 5 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 16 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 17 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 18 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 20 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LENGTH</td>
<td>291.82</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 51 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 1 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 2 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 3 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 4 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 5 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 20 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LENGTH</td>
<td>297.06</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 61 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 1 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 2 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 3 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 4 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 5 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 26 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 27 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 28 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 29 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( 30 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LENGTH</td>
<td>210.26</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F-VALUES</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NAMN</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 1 )</td>
<td>906.00</td>
</tr>
<tr>
<td>( 7 )</td>
<td>58.00</td>
</tr>
<tr>
<td>( 11 )</td>
<td>56.00</td>
</tr>
<tr>
<td>( 4 )</td>
<td>15.00</td>
</tr>
<tr>
<td>( 5 )</td>
<td>32.00</td>
</tr>
<tr>
<td>( 7 )</td>
<td>44.00</td>
</tr>
<tr>
<td>( 3 )</td>
<td>42.00</td>
</tr>
<tr>
<td>( 8 )</td>
<td>37.00</td>
</tr>
<tr>
<td>( 9 )</td>
<td>17.00</td>
</tr>
<tr>
<td>( 10 )</td>
<td>17.00</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>( 60 )</td>
<td>294.00</td>
</tr>
<tr>
<td>( 61 )</td>
<td>140.00</td>
</tr>
<tr>
<td>( 62 )</td>
<td>514.00</td>
</tr>
<tr>
<td>( 63 )</td>
<td>437.00</td>
</tr>
<tr>
<td>( 64 )</td>
<td>1042.00</td>
</tr>
<tr>
<td>( 65 )</td>
<td>277.00</td>
</tr>
<tr>
<td>( 66 )</td>
<td>8095.00</td>
</tr>
<tr>
<td>( 67 )</td>
<td>15884.00</td>
</tr>
</tbody>
</table>

*continued*
PENNSYLVANIA POPULATION DENSITY

MAP SIZE 16.00 INCHES LONG BY 16.00 INCHES WIDE
SHEETS OF PAPER IS 15
6 SHEETS ARE REQUIRED TO DRAW 1
25.00 25.00 50.00
3 100.00 100.00 100.00
10 MAP TEXT
164 PERCENT
0.00 CMER
16 MAP SCALE IS 0.11
16 MAP LEGENDS ARE 1.50 AT TOP, 1.60 AT LEFT, 2.00 AT BOTTOM, AND 1.50 AT RIGHT
TIME = 13:10:
REQUEST TIME = 11.56 SECONDS FOR INPUT

continued
MAP 1

C. POPULATION DENSITY
C. DENSITIES PER SQUARE MILE

MAP SCALE = 0.1100 INCHES ON INPUT MAP/UNITS ON OUTPUT MAP

MAP SHOULD BE PRINTED AT 5.0 PPM PER INCH AND 10.2 COLUMNS PER INCH

ROW = (DOWN CORRELATION) = -0.82) * 0.4400
COLUMN = (ACROSS CORRELATION) = -0.14) * 1.1000

CONFIRMED INF'S PER MAP

<table>
<thead>
<tr>
<th>ZONE</th>
<th>TYPE</th>
<th>ROW</th>
<th>COLUMN</th>
<th>DATING</th>
<th>VALUE</th>
<th>LEVEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>A</td>
<td>31</td>
<td>42</td>
<td>1</td>
<td>109.30</td>
<td>6</td>
</tr>
<tr>
<td>23</td>
<td>A</td>
<td>13</td>
<td>88</td>
<td>2</td>
<td>50.00</td>
<td>2</td>
</tr>
<tr>
<td>31</td>
<td>A</td>
<td>10</td>
<td>102</td>
<td>3</td>
<td>45.00</td>
<td>2</td>
</tr>
<tr>
<td>41</td>
<td>A</td>
<td>17</td>
<td>175</td>
<td>4</td>
<td>15.00</td>
<td>1</td>
</tr>
<tr>
<td>51</td>
<td>A</td>
<td>56</td>
<td>243</td>
<td>6</td>
<td>46.00</td>
<td>2</td>
</tr>
<tr>
<td>61</td>
<td>A</td>
<td>39</td>
<td>794</td>
<td>7</td>
<td>40.00</td>
<td>2</td>
</tr>
<tr>
<td>71</td>
<td>A</td>
<td>67</td>
<td>119</td>
<td>8</td>
<td>39.00</td>
<td>2</td>
</tr>
<tr>
<td>82</td>
<td>A</td>
<td>64</td>
<td>125</td>
<td>9</td>
<td>17.00</td>
<td>1</td>
</tr>
<tr>
<td>93</td>
<td>A</td>
<td>57</td>
<td>214</td>
<td>10</td>
<td>517.00</td>
<td>8</td>
</tr>
<tr>
<td>103</td>
<td>A</td>
<td>115</td>
<td>257</td>
<td>9</td>
<td>234.00</td>
<td>6</td>
</tr>
<tr>
<td>113</td>
<td>X</td>
<td>29</td>
<td>275</td>
<td>6</td>
<td>120.00</td>
<td>6</td>
</tr>
<tr>
<td>121</td>
<td>X</td>
<td>98</td>
<td>289</td>
<td>4</td>
<td>694.00</td>
<td>9</td>
</tr>
<tr>
<td>131</td>
<td>X</td>
<td>48</td>
<td>286</td>
<td>4</td>
<td>694.00</td>
<td>9</td>
</tr>
<tr>
<td>141</td>
<td>X</td>
<td>106</td>
<td>212</td>
<td>6</td>
<td>1122.00</td>
<td>10</td>
</tr>
<tr>
<td>151</td>
<td>X</td>
<td>117</td>
<td>294</td>
<td>4</td>
<td>577.00</td>
<td>6</td>
</tr>
<tr>
<td>161</td>
<td>X</td>
<td>112</td>
<td>318</td>
<td>5</td>
<td>800.00</td>
<td>10</td>
</tr>
<tr>
<td>171</td>
<td>X</td>
<td>116</td>
<td>326</td>
<td>6</td>
<td>1554.00</td>
<td>0</td>
</tr>
</tbody>
</table>

TIME = 14:14:40.20

CLASS TIME = 1.25 sec. 435 sec. for initial calculations

continued
**EDUCATIONAL INFORMATION NETWORK**

TIME = 18:19:47.4
FLAPSE TIME = 0.76 SECONDS FOR HISTOGRAM

C PENNSYLVANIA POPULATION DENSITY
C PERSONS PER SQUARE MILE

1965 DATA
C. E. GERLACH

DATA VALUE EXTREMES ARE 11.00 150.00

**ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL**

<table>
<thead>
<tr>
<th>LEVEL</th>
<th>MINIMUM</th>
<th>MAXIMUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11.00</td>
<td>35.93</td>
</tr>
<tr>
<td>2</td>
<td>35.93</td>
<td>61.76</td>
</tr>
<tr>
<td>3</td>
<td>61.76</td>
<td>113.93</td>
</tr>
<tr>
<td>4</td>
<td>113.93</td>
<td>210.76</td>
</tr>
<tr>
<td>5</td>
<td>210.76</td>
<td>410.72</td>
</tr>
<tr>
<td>6</td>
<td>410.72</td>
<td>510.65</td>
</tr>
<tr>
<td>7</td>
<td>510.65</td>
<td>613.58</td>
</tr>
<tr>
<td>8</td>
<td>613.58</td>
<td>1010.29</td>
</tr>
</tbody>
</table>

**PERCENTAGE OF TOTAL ABSOLUTE VALUE RANGE APPLYING TO EACH LEVEL**

<table>
<thead>
<tr>
<th>LEVEL</th>
<th>PERCENTAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.16</td>
</tr>
<tr>
<td>2</td>
<td>0.16</td>
</tr>
<tr>
<td>3</td>
<td>0.12</td>
</tr>
<tr>
<td>4</td>
<td>0.04</td>
</tr>
<tr>
<td>5</td>
<td>0.32</td>
</tr>
<tr>
<td>6</td>
<td>0.16</td>
</tr>
<tr>
<td>7</td>
<td>0.64</td>
</tr>
<tr>
<td>8</td>
<td>2.57</td>
</tr>
<tr>
<td>9</td>
<td>0.64</td>
</tr>
<tr>
<td>10</td>
<td>63.49</td>
</tr>
</tbody>
</table>

**FREQUENCY DISTRIBUTION OF DATA PRINT VALUES IN EACH LEVEL**

<table>
<thead>
<tr>
<th>LEVEL</th>
<th>SYMBOLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>--------</td>
</tr>
<tr>
<td>2</td>
<td>--------</td>
</tr>
<tr>
<td>3</td>
<td>--------</td>
</tr>
<tr>
<td>4</td>
<td>--------</td>
</tr>
<tr>
<td>5</td>
<td>--------</td>
</tr>
<tr>
<td>6</td>
<td>--------</td>
</tr>
<tr>
<td>7</td>
<td>--------</td>
</tr>
<tr>
<td>8</td>
<td>--------</td>
</tr>
<tr>
<td>9</td>
<td>--------</td>
</tr>
<tr>
<td>10</td>
<td>--------</td>
</tr>
</tbody>
</table>

**FREQUENCY DISTRIBUTION**

<table>
<thead>
<tr>
<th>LEVEL</th>
<th>SYMBOLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>--------</td>
</tr>
<tr>
<td>2</td>
<td>--------</td>
</tr>
<tr>
<td>3</td>
<td>--------</td>
</tr>
<tr>
<td>4</td>
<td>--------</td>
</tr>
<tr>
<td>5</td>
<td>--------</td>
</tr>
<tr>
<td>6</td>
<td>--------</td>
</tr>
<tr>
<td>7</td>
<td>--------</td>
</tr>
<tr>
<td>8</td>
<td>--------</td>
</tr>
<tr>
<td>9</td>
<td>--------</td>
</tr>
<tr>
<td>10</td>
<td>--------</td>
</tr>
</tbody>
</table>

TIME = 18:19:47.48
FLAPSE TIME = 0.76 SECONDS FOR HISTOGRAM
COST ESTIMATE

For the jobs listed on the Sample Input, the total running time was 118 seconds for the contour map and 168 seconds for the conformant map. At the current rate for computer time ($0.11/sec.) at The Pennsylvania State University, the chargeable time amounts are the following.

Charge to user = computer time + network overhead
   = $12.98 + network overhead— for the contour map
   = $18.48 + network overhead— for the conformant map

CONTENTS—SYMAP

1- 3   Identification & Abstract
5- 7   User Instructions
9-26   I/O
27     Cost—Contents
Quick-Draw Graphics System

QDGS

The Pennsylvania State University Computation Center


FORTRAN IV

IBM System 360/67

Decks and listings presently available

Daniel L. Bernitt, 105 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

QDGS is a set of FORTRAN subroutines that can be used to draw charts, graphs, diagrams, maps, or any other form of pictorial output. The graphic material is produced on the CalComp plotter and other graphic devices as they become available.

To use QDGS, a program is written in FORTRAN IV for the IBM System/360 computer. This program calls the QDGS subroutines; the subroutines produce output (in the form of cards or tape) that is later put into a special system to produce the actual graphical output. When the cards (or tape) are received from the dispatcher and the user has verified that the run was successful, he usually submits the cards to the dispatcher along with a plot request form.

The three concepts underlying the design of QDGS are (1) the description of an elementary picture in terms of coordinates, (2) the geometric transformation of pictures (into various positions and orientations), and (3) the building of complex pictures from elementary ones. There will be subroutines to do each of the three basic processes. In addition, there are special routines to draw axes for graphs and charts, to do lettering and numbering, and automatically to do the necessary task of punching the cards or
writing the tape used in plotting (physically drawing on the plotting device). QDGS is intended as a nucleus about which the user's graphic routines, tailored to his own unique application, can be written.
USER INSTRUCTIONS

Initialization

Before any portion of the QDGS may be used by a program, four arrays must be specified. These arrays are used by the system as a working area and may not be used or altered in any way by the user's program. It is difficult to estimate the size of the arrays as space requirements vary widely with different program; however, the following declarations will do for many jobs. If the message LAVS EXHAUSTED appears, the 1000 may be increased.

REAL D(6,1000)
INTEGER N 2 A(1000), B(1000), C(1000)

Any FORTRAN variable names may be used for the A, B, C, and D. The first subscript of the D array must be 6. Once the arrays have been initialized, and before any other QDGS subroutines are called, the following statement must be used

CALL INITQ(A,B,C,D,1000)

Defining Elementary Pictures

Pictures are defined in Cartesian coordinates; however, with each point (x,y), we now associate a third coordinate z, which has a code for the condition of the plotter pen in arriving at the point (x,y). The convention used is to code z=0 to indicate "without drawing" and z=1 to indicate "drawing." The desired picture is achieved, therefore, by storing these ordered triples. This is the purpose of the ADPTQ subroutine (see Summary of Subroutines).

Building Pictures from Pictures

In the QDGS, pictures are formed out of line segments defined with ADPTQ and by transformed versions of pictures previously defined. A geometric transformation is applied to the previously defined picture before being placed into another picture. There are six basic transformations (and compounds of these) allowed by the QDGS:

1. translation in X (TX)—motion of a figure horizontally
2. translation in Y (TY)—motion of a figure vertically
3. scaling in S (SX)—horizontal expansion or contraction
4. scaling in Y (SY)—vertical expansion or contraction
5. rotation (TH)—the TH is for theta, traditionally used to denote an angle—here measured counterclockwise in radians

continued
(6) skew (or shear) (SK)—a tilting that leaves height unchanged, measure in radians clockwise from the positive Y direction.

Transformations are represented by variables and must, therefore, be dimensioned (using 6 as the dimension); their values, however, are set by calling the subroutine XFSQ. Compound and inverse transformations may be performed by using subroutines XFXQ and XFINQ, respectively. Adding pictures (building more complex ones) can be achieved by using ADELQ, which also allows a picture to be transformed before being added to another picture.

Displaying a Completed Picture

Once a picture has been created and given a name, the card or tape output representing it may be obtained. Up to now, pictures have been described in terms of abstract coordinates, but the physical size is determined by a physical scale factor in the call to the outputting routine, subroutine DISPQ. The scale factor itself, however, is determined by the desired units of the user and the plotter's raster unit. (A raster unit is the smallest distance over which a line can be drawn by a particular device.)

Axes

A single subroutine, AXISQ, provides a logarithmic or linear axis with appropriate labeling of the tic marks as an option.

Drawing Labels and Titles

A legend on a graph or chart, the text that accompanies a drawing, sizes, tolerance, and the like are produced by subroutine LABELQ (for alphanumeric) and NBRQ (for numbers). A label is itself a picture and differs from other pictures only in that a subroutine sets up the picture automatically. The user transforms it to the desired size and position by using the same transformations as for any other picture.

REFERENCES

### Summary of Subroutines

<table>
<thead>
<tr>
<th>Name</th>
<th>Form of Call</th>
<th>Meaning of Parameters</th>
<th>Purpose and/or Abilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>INITQ</td>
<td>CALL INITQ (A,B,C,D,1000)</td>
<td>any FORTRAN variables names for A,B,C,D</td>
<td>initialization (required after declaring A,B,C,D)</td>
</tr>
<tr>
<td>ADPTQ</td>
<td>CALL ADPTQ (X,Y,Z,NAME)</td>
<td>X,Y: Cartesian coordinates; Z: plotter per code NAME: name for picture</td>
<td>adds (stores) points to an elementary picture</td>
</tr>
<tr>
<td>XFSTQ</td>
<td>CALL XFSTQ (TX,TY,SX,SY,TH,SK,Q)</td>
<td>Q: first six already described; transformation to be given the value</td>
<td>transforming routine (transformations performed in order SK,TH,SY,SX,TY,TX)</td>
</tr>
<tr>
<td>XFXFQ</td>
<td>CALL XFXFQ (S,Q,T)</td>
<td>Q,S: two predefined transformations T: the new compound transformation of Q and S</td>
<td>allows compounding of transformations (Q performed first, then S)</td>
</tr>
<tr>
<td>XFINQ</td>
<td>CALL XFINQ (Q,QI)</td>
<td>Q: predefined transformation new, inverse transformations QI:</td>
<td>creates inverse transformations</td>
</tr>
<tr>
<td>ADELQ</td>
<td>CALL ADELQ (N,Q,M)</td>
<td>N: picture to be added Q: transformation to be applied to N M: new picture or picture to be built up</td>
<td>allows pictures of any structure depth to be created</td>
</tr>
<tr>
<td>DISPQ</td>
<td>CALL DISPQ (N,SCLF)</td>
<td>N: name of picture to be output SCLF: scale factor for plotting</td>
<td>actual displaying routine; for various plotting devices</td>
</tr>
</tbody>
</table>

*continued*
### Summary of Subroutines (cont.)

<table>
<thead>
<tr>
<th>Name</th>
<th>Form of Call</th>
<th>Meaning of Parameters</th>
<th>Purpose and/or Abilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>AXIQ</td>
<td>CALL AXIQ (SIZE, K, TMIN, TINC, HITE, N, Q, M)</td>
<td>SIZE: overall axis length</td>
<td>Sp,e options may be indicated by using each parameter. Minus signs before the values of SIZE, K, and HITE indicate axis suppression, logarithmic scaling, and placement of numbers on the other side of an axis. If K is negative, TINC represents the constant ratio between marks rather than the linear increment; if N=-2, numbering is suppressed for the axis (TMIN and TINC are ignored)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>K: number of segments</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TMIN: value placed at first mark</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>TINC: increment for each succeeding mark</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HITE: size of axis marks and numbers appearing with the axis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>N: number of digits to right of decimal point in numbers</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Q: transformation to be applied to axis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>M: name of axis pictured</td>
<td></td>
</tr>
<tr>
<td>LABLQ</td>
<td>CALL LABLQ (LABEL, LGTH, Q, M)</td>
<td>LABEL: label (in quotes)</td>
<td>LABEL may be replaced by a variable representing alphanumeric data; a minus sign before LGTH indicates centering rather than left-justification</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LGTH: number of characters</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Q: transformation to be applied</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>N: picture to which label is to be added</td>
<td></td>
</tr>
<tr>
<td>NMBRQ</td>
<td>CALL NMBRQ (R, N, IC, Q, M)</td>
<td>R: real number desired</td>
<td>N=-1 suppresses decimal</td>
</tr>
<tr>
<td></td>
<td></td>
<td>N: number of digits behind decimal point</td>
<td>IC=0 for centering, 1 for left-justification; R may have no more than 8 digits to the left and 7 digits to the right of the decimal point</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IC: position code</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Q, M: same as above</td>
<td></td>
</tr>
</tbody>
</table>
SAMPLE INPUT WITH CORRESPONDING OUTPUT

EXAMPLE OF A COMPLETE PROGRAM WITH OUTPUT

PROGRAM TO PRODUCE FIGURE 8 FOR THE QDGS PRIMER

INITIALIZATION
INTEGER*2 A(1000),B(1000),C(1000)
INTEGER M/0/
REAL D(6,1000),Q(6),PI/3.141593/
CALL INTQ(A,B,C,D,1000)

CREATE TRANSFORM FOR Y-AXIS
CALL XFSTQ(0.0,-1.5,1.00,1.00,PI/2.0,0.0,Q)
CALL AXISQ(3.0,0,-.3,.1,.1,.08,.1,Q,M)

CREATE TRANSFORM FOR THE X-AXIS AND ADD AXIS TO M
CALL XFSTQ(0.0,0.0,1.00,1.00,0.0,0.0,Q)
CALL AXISQ(3.0,-.3,1.0,10.0,0.0,.08,-.1,Q,M)

PREPARE A TRANSFORM FOR THE LABEL
CALL XFSTQ(1.5,-.75,.12,.13,0,.3,Q)
ADD THE LABEL TO M
CALL LABLQ('FIGURE 8',-8,Q,M)

DISPLAY THE COMPLETED FIGURE 8, SCALED FOR THE CALCOMP
CALL DISPQ(M,200.0)

NOTE: ALL OF THE ILLUSTRATIONS FOR THE QDGS PRIMER WERE
DRAWN ON THE CALCOMP PLOTTER. ONLY THOSE GRAPHIC
SUBROUTINES MENTIONED IN THIS PRIMER WERE USED.
STOP
END

FIGURE 8

continued
PROGRAM TO READ POINTS AND PLOT A GRAPH

ASSUME THE RANGES OF X AND Y ARE KNOWN
X: -5 TO +5
Y: -2 TO +2
AND THE (X,Y) PAIRS HAVE BEEN SORTED IN ORDER OF INCREASING X.

CALL THE GRAPH "GRAPH"

INTEGER GRAPH/0/,PLUS/0/,INPUT/5/

INITIALIZE

INTEGER*2 AA(1000),BB(1000),CC(1000)
REAL DD(6,1000),PI/3.141593/,Q(6)

CALL INITNAA,BB,CC,DD,1000)

PUT X-AXIS ON GRAPH
MAKE AXIS RUN FROM -5 TO +5 WITH 10 SEGMENTS, THUS 'SIZE' IS 10. AND 'K' IS 10. AXIS PROVIDES AN AXIS STARTING AT 0 AND STRETCHING IN THE POSITIVE X-DIRECTION. THEREFORE, SET TX=-5.

IN ORDER TO START THIS AXIS AT -5. NUMBER EACH TIC MARK STARTING WITH -5 AT THE LEFT END AND INCREMENTING BY 1. SUPPRESS THE DECIMAL POINT('N=-1) AND MAKE THE DIGITS .1 UNIT HIGH('HITE'=1).

CALL XFSTO(-5.,01.0.,0.,0.,Q)
CALL AXISQ(10.,10, -5.,1.,.1,-1,Q,GRAPH)

LABEL X-AXIS
MAKE LETTERS .1 HIGH TO MATCH NUMBERS. USE N=-10 TO CENTRE LABEL AT (4.,.5)

CALL XFSTO(4 ,.5,.1,.1,0.,0.,Q)
CALL LABLO('X-VARIABLE',-10,0,GRAPH)

PUT Y-AXIS ON GRAPH
USE 'SIZE'=4. THIS GIVES AN AXIS FROM 0 TO 4 IN X-DIRECTION. ROTATE BY PI/2 FOR AXIS FROM 0 TO 4 IN Y-DIRECTION. TRANSLATE BY -2. FOR AXIS FROM -2 TO +2 IN Y-DIRECTION. NUMBER TIC MARKS AS BEFORE BUT USE 'HITE'=1 TO PUT TIC MARKS AND NUMBERS ON THE OTHER SIDE.

CALL XFSTO(0.,-2.0.0.,PI/2.,0.,Q)
CALL AXISQ(4.,4,-2.0.,.1,-1,Q,GRAPH)

LABEL Y-AXIS
MAKE LETTERS .1 HIGH TO MATCH NUMBERS. USE N=-10 TO CENTRE LABEL AT (0.4,1.)

CALL XFSTO(-.4,1.1,.1,PI/2.,0.,Q)
CALL LABLO('Y-VARIABLE',-10,Q,GRAPH)

IF DATA POINTS ARE TO BE DENOTED BY A SPECIAL SYMBOL, CREATE IT NOW,E.G.,A PLUS SIGN CALLED "PLUS"

MAKE IT SO THAT THE PEN IS BACK AT THE STARTING POINT AFTER DRAWING IT. ALSO MAKE IT A STANDARD SIZE(1 BY 1).

CALL ADPTQ (.5,0.,PLUS)
CALL ADPT0(0.,-.5,0,PLUS)
CALL ADPT0(0.,.5,1,PLUS)
CALL ADPT0(0.,0.,0,PLUS)

MOVE PEN TO FIRST DATA POINT WITHOUT DRAWING A LINE.

READ(INPUT,2,END=4) X,Y
2 FORMAT(2F10.2)
CALL ADPT0(X,Y,0,GRAPH)

OMIT NEXT SET OF STATEMENTS IF NO SPECIAL SYMBOL IS TO BE PLACED AT THIS POINT.

MAKE PLUS SIGN .1 UNIT SQUARE ON GRAPH,PLACED AT (X,Y)
CALL XFSTO(X,Y,1.1,0.,0.,Q)
CALL ADLO(PLUS,Q,GRAPH)

READ DATA POINTS AND PUT ON GRAPH

3 READ(INPUT,2,END=4) X,Y
OMIT NEXT STATEMENT IF POINTS ARE NOT TO BE JOINED BY A LINE.

CALL ADPT0(X,Y,1,GRAPH)

OMIT NEXT SET OF STATEMENTS IF NO SPECIAL SYMBOL IS TO BE PLACED AT EACH POINT.

CALL XFSTO(X,Y,1.1,0.,0.,Q)
CALL ADLO(PLUS,Q,GRAPH)

(X,Y) HAS NOW BEEN ADDED TO GRAPH

90 TO 3

4 CONTINUE

GRAPH IS COMPLETED. CHOOSE SCALEFACTOR AND DISPLAY IT.

CALL DISPO(GRAPH,200.)
STOP
END
COST ESTIMATE

For the jobs listed as the Sample Input, no exact timing figures were available. However, none of the included examples would require more than 15 seconds of computer time to run. At the current rate of charge at The Pennsylvania State University ($0.11/sec.), this amounts to a maximum of $1.65 per example.

Approximate charge to user = $1.65 (maximum) \times \text{number of examples} + \text{network overhead}
The KGIC program was written to facilitate the analysis of the environmental distribution of graphic characters. It produces a KWIC-like listing of all occurrences of a given grapheme along with the graphic environment in which each instance appeared. The listing may be sorted either forward or backward from the key grapheme to facilitate inspection. Provision is made for specification by the user of special alphabets for foreign languages or for phonemic transcriptions.

If any word appears more than once in the data to be processed, only one set or records will be produced for that word. A counter will be increased and the frequency of occurrence printed in both the alphabetical listing and in the KGIC listing. The alpha listing thus will contain a complete frequency count of the corpus under consideration.

It should be noted that whenever the user can attribute phonetic or phonemic status to individual graphemes, the KGIC listing provides correspondingly significant information about phonetic and/or phonemic environments.
The program produces the items listed below as output.

1. an alphabetical listing of all words processed by the KGIC program, along with their absolute and relative frequencies of occurrence,

2. the KGIC listing itself grouped by alphabetical character, with the absolute frequency of occurrence given for each unique occurrence, and the total number of occurrences and the total number of unique occurrences given for each character,

3. a summary table containing the absolute and relative frequencies of occurrence for both the total number of occurrences and for the total of unique occurrences,

4. optionally, a horizontal bar graph of the relative frequencies of all occurrences of each grapheme,

5. a number of summary statistics, i.e.,
   a) total number of words processed, i.e., tokens
   b) average length of word
   c) total number of unique words, i.e., types
   d) the type/token ratio
   e) total number of characters processed
   f) total number of unique occurrences of all characters

6. a statement of all program options used in a particular run, and a complete listing of the EMICTT, defining the alphabet in use for that run.
USER INSTRUCTIONS

A number of the run-time options are available. One or more options may be selected by including the appropriate information in the data set OPTIN. For each available option, except TITLE, a default is indicated which will apply if the given option is not specified. Two or more options are separated by commas, and the final option is followed by a semicolon.

1. TITLE
The user must specify a project title to be used in the headings of certain parts of the program output. The title must be no longer than 80 characters, including blanks, and must be enclosed within single quote marks, as follows,

   TITLE = 'users project title'

2. SORT
As indicated previously, the KGIC listing can be sorted either forward or backward from the key character. Merely code BACKWARD or FORWARD, as follows,

   SORT = 'BACKWARD'
   'FORWARD'

   If SORT is not specified, the list is alphabetized forward by default.

3. GRAPHX
A horizontal bar graph displaying graphically the frequency of occurrence of each grapheme which appeared in the data may be produced. Code,

   GRAPHX = 'YES'
     'NO'

   The default option bypasses graphing.

4. SPACING
To indicate desired spacing of the alpha and KGIC listings, merely code

   SPACING = n

   where n is the number of lines skipped after printing. Default is 1 (i.e., single spacing).

   continued
5. EMICTT

Because the program is designed for graphic analysis of foreign languages as well as English, provision has been made for the specification of special alphabets designed by the user for foreign languages or for phonetic transcription. Such specification is done through the Expanded Master Input Character Translate Table (EMICTT), which consists of two major parts. The first, MICTT, specifies the use to be made of each of the 256 characters recognized by the 360. The codes employed are as follows.

1 = Ignore, not used as print.
2 = Delimiter for ID fields, comments, titles, and other information to be ignored in processing.
3 = Space.
4 = Character to be considered part of the alphabet in use.

If any character other than 1 to 4 is coded, the improper code will be replaced by code 4, and an error message produced.

The second part of EMICTT specifies the character(s) to be produced on output for each character designated as part of the alphabet (i.e., those coded '4'). There are two possibilities. First, each character can be represented by a single character from the S/360 TN train, or other available print train. Second, for additional flexibility, two characters may be overprinted to yield a composite character. For example, the break character ('\') might be used to distinguish low toned vowels from other vowels. In either case, printing is determined by specification of the appropriate character in either PRT(1), for base characters, or PRT(2), for overprinted characters.

For both MICTT and PRT(1) and PRT(2), default values specified are for the standard English (ROMAN) alphabet, as given in Appendix A. Changes are made in MICTT by coding as follows,

MICTT (xxx) = n

continued
where \(xxx\) is the decimal number from 1 to 255 specifying the internal character under consideration, and \(n\) is the usage code (1-4). Changes are made in the print routine by coding

\[
PRT(i, xxx) = 'c'
\]

where \(xxx\) is the decimal number from 1 to 255, \(i\) designates the portion of \(PRT\) involved (i.e., \(1 = \text{base character}, 2 = \text{overprint}\)), and \(c\) is the punched code for the desired print character.

Note that hex zero is a reserved character used in the production of the alpha listing. Attempts to change \(MICTT(0)\) or \(PRT(i,0)\) will be ignored and an error message produced.

Multiple changes in \(PRT\) and \(MICTT\) can be made by separating them by commas.

For example, to treat the hyphen (-) and the apostrophe ('') as characters to be ignored rather than part of the alphabet, code as follows,

\[
MICTT(96) = 1, PRT(1,96) = '\',
\]
\[
MICTT(125) = 1, PRT(1,125) = '\',
\]

A modification of \(EMICTT\) for use with a phonemic transcription of the Mayan Language is shown in Appendix B. Upper-case consonants are used for glottalized sounds. High-, mid-, and low-toned vowels are represented by upper-case, lower-case and underlined lower-case vowels respectively.

**Description of Input**

Input consists of two files, one carrying the run-time program options (OPTIN), and the other carrying the data itself (SYSIN).

File OPTIN consists of one or more cards containing the user-specified project title and one or more program options, or none at all. The title and each of the program options are separated from each other by commas, and the final option is followed by a semicolon. If no program options are specified the title must be followed by a semicolon.
A typical example might be as follows,

```
//GO.OPTIN DD *
TITLE = 'MAYAN PROJECT', SPACING = 2
GRAPHX = 'YES';
/*
```

File SYSIN carries the data to be analyzed. The input stream may contain any acceptable 8-bit code except 00000000 (hex 00) which is reserved for internal use. For example,

```
//GO.SYSIN DD *
THIS IS A SAMPLE OF INPUT DATA.
```

Running Instructions

The KGIC load module is contained in a partitioned data set DS09810.JALIB, stored in disk, and can thus be executed directly without compilation. In addition to the data input file, SYSIN, and the program options file, OPTIN, the program requires three work files for the SORT routine and the file SYSPRINT which contains the program output.

The JCL for a typical run is illustrated below. This is only an illustration, however, and must be adapted to the user's specifications.

```
---current OS JOB card---
//JOBLIB DD DSN=DS09810.JALIB,DISP=SHR
// EXEC PGM=KGIC
//SORTWK01 DD UNIT=SYSDA,SPACE=(TRK,500,,CONTIG)
//SORTWK02 DD UNIT=(SYSDA,SEP=SORTWK01),SPACE=(TRK,500,,CONTIG)
//SORTWK03 DD UNIT=(SYSDA,SEP=(SORTWK02,SORTWK01)),
//  SPACE=(TRK,500,,CONTIG)
//SYSOUT DD SYSOUT=A
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
//SYSLIN DD UNIT=SYSDA,SPACE=(TRK,(10,10))
//SYSLUTI DD UNIT=SYSDA,SPACE=(TRK,(10,10))
//SYSPRINT DD SYSOUT=A CONTAINS THE PROGRAM OUTPUT
//SYSIN DD UNIT=TAPE9,DISP=(OLD,KEEP) CONTAINS INPUT DATA
//OPTIN DD *
---One of more cards with program options and project title---
```

continued
**Appendix A: Default values for EMICTT**

**EXPANDED MASTER INPUT CHARACTER TRANSLATE TABLE**

<table>
<thead>
<tr>
<th>DEC</th>
<th>CODE</th>
<th>PRT(1)</th>
<th>PRT(2)</th>
<th>DEC</th>
<th>CODE</th>
<th>PRT(1)</th>
<th>PRT(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td>41</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>42</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td>43</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td>44</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td></td>
<td></td>
<td>45</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td></td>
<td></td>
<td>46</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td></td>
<td></td>
<td>47</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td></td>
<td></td>
<td>48</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td></td>
<td></td>
<td>49</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td></td>
<td></td>
<td>50</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td></td>
<td></td>
<td>51</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td></td>
<td></td>
<td>52</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td></td>
<td></td>
<td>53</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td></td>
<td></td>
<td>54</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td></td>
<td></td>
<td>55</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td></td>
<td></td>
<td>56</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td></td>
<td></td>
<td>57</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td></td>
<td></td>
<td>58</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td></td>
<td></td>
<td>59</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>1</td>
<td></td>
<td></td>
<td>60</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td></td>
<td></td>
<td>61</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>1</td>
<td></td>
<td></td>
<td>62</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>1</td>
<td></td>
<td></td>
<td>63</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>1</td>
<td></td>
<td></td>
<td>64</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td></td>
<td></td>
<td>65</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td></td>
<td></td>
<td>66</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>1</td>
<td></td>
<td></td>
<td>67</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>1</td>
<td></td>
<td></td>
<td>68</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>1</td>
<td></td>
<td></td>
<td>69</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>1</td>
<td></td>
<td></td>
<td>70</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>1</td>
<td></td>
<td></td>
<td>71</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>1</td>
<td></td>
<td></td>
<td>72</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>1</td>
<td></td>
<td></td>
<td>73</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>1</td>
<td></td>
<td></td>
<td>74</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>1</td>
<td></td>
<td></td>
<td>75</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>1</td>
<td></td>
<td></td>
<td>76</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>1</td>
<td></td>
<td></td>
<td>77</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>1</td>
<td></td>
<td></td>
<td>78</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>1</td>
<td></td>
<td></td>
<td>79</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>1</td>
<td></td>
<td></td>
<td>80</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEC</td>
<td>CODE</td>
<td>PRT(1)</td>
<td>PRT(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>------</td>
<td>--------</td>
<td>--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>81</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>82</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>83</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>84</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>85</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>86</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>87</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>88</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>89</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>91</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>92</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>93</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>94</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>95</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>96</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>97</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>98</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>99</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>101</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>102</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>103</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>104</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>105</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>106</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>107</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>108</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>109</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>110</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>111</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>112</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>113</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>114</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>115</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>116</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>117</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>118</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>119</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>120</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>121</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>122</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>123</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>124</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>125</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DEC</th>
<th>CODE</th>
<th>PRT(1)</th>
<th>PRT(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>126</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>127</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>128</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>129</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>130</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>131</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>132</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>133</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>134</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>135</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>136</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>137</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>138</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>139</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>140</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>141</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>142</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>143</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>144</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>145</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>146</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>147</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>148</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>149</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>150</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>151</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>152</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>153</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>154</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>155</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>156</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>157</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>158</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>159</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>161</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>162</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>163</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>164</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>165</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>166</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>167</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>168</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>169</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>170</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th>DEC</th>
<th>CODE</th>
<th>PRT(1)</th>
<th>PRT(2)</th>
<th>DEC</th>
<th>CODE</th>
<th>PRT(1)</th>
<th>PRT(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>171</td>
<td>1</td>
<td></td>
<td></td>
<td>216</td>
<td>4</td>
<td>Q</td>
<td></td>
</tr>
<tr>
<td>172</td>
<td>1</td>
<td></td>
<td></td>
<td>217</td>
<td>4</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>173</td>
<td>1</td>
<td></td>
<td></td>
<td>218</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>174</td>
<td>1</td>
<td></td>
<td></td>
<td>219</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>175</td>
<td>1</td>
<td></td>
<td></td>
<td>220</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>176</td>
<td>1</td>
<td></td>
<td></td>
<td>221</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>177</td>
<td>1</td>
<td></td>
<td></td>
<td>222</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>178</td>
<td>1</td>
<td></td>
<td></td>
<td>223</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>179</td>
<td>1</td>
<td></td>
<td></td>
<td>224</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>180</td>
<td>1</td>
<td></td>
<td></td>
<td>225</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>181</td>
<td>1</td>
<td></td>
<td></td>
<td>226</td>
<td>4</td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>182</td>
<td>1</td>
<td></td>
<td></td>
<td>227</td>
<td>4</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>183</td>
<td>1</td>
<td></td>
<td></td>
<td>228</td>
<td>4</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>184</td>
<td>1</td>
<td></td>
<td></td>
<td>229</td>
<td>4</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>185</td>
<td>1</td>
<td></td>
<td></td>
<td>230</td>
<td>4</td>
<td>W</td>
<td></td>
</tr>
<tr>
<td>186</td>
<td>1</td>
<td></td>
<td></td>
<td>231</td>
<td>4</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>187</td>
<td>1</td>
<td></td>
<td></td>
<td>232</td>
<td>4</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>188</td>
<td>1</td>
<td></td>
<td></td>
<td>233</td>
<td>4</td>
<td>Z</td>
<td></td>
</tr>
<tr>
<td>189</td>
<td>1</td>
<td></td>
<td></td>
<td>234</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>190</td>
<td>1</td>
<td></td>
<td></td>
<td>235</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>191</td>
<td>1</td>
<td></td>
<td></td>
<td>236</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>192</td>
<td>1</td>
<td></td>
<td></td>
<td>237</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>193</td>
<td>4</td>
<td>A</td>
<td></td>
<td>238</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>194</td>
<td>4</td>
<td>B</td>
<td></td>
<td>239</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>195</td>
<td>4</td>
<td>C</td>
<td></td>
<td>240</td>
<td>4</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>196</td>
<td>4</td>
<td>D</td>
<td></td>
<td>241</td>
<td>4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>197</td>
<td>4</td>
<td>E</td>
<td></td>
<td>242</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>198</td>
<td>4</td>
<td>F</td>
<td></td>
<td>243</td>
<td>4</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>199</td>
<td>4</td>
<td>G</td>
<td></td>
<td>244</td>
<td>4</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>4</td>
<td>H</td>
<td></td>
<td>245</td>
<td>4</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>201</td>
<td>4</td>
<td>I</td>
<td></td>
<td>246</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>202</td>
<td>1</td>
<td></td>
<td></td>
<td>247</td>
<td>4</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>203</td>
<td>1</td>
<td></td>
<td></td>
<td>248</td>
<td>4</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>204</td>
<td>1</td>
<td></td>
<td></td>
<td>249</td>
<td>4</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>205</td>
<td>1</td>
<td></td>
<td></td>
<td>250</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>206</td>
<td>1</td>
<td></td>
<td></td>
<td>251</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>207</td>
<td>1</td>
<td></td>
<td></td>
<td>252</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>208</td>
<td>1</td>
<td></td>
<td></td>
<td>253</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>209</td>
<td>4</td>
<td>J</td>
<td></td>
<td>254</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>210</td>
<td>4</td>
<td>K</td>
<td></td>
<td>255</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>211</td>
<td>4</td>
<td>L</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>212</td>
<td>4</td>
<td>M</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>213</td>
<td>4</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>214</td>
<td>4</td>
<td>O</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>215</td>
<td>4</td>
<td>P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

continued
### Appendix B: EMICTT for a Mayan Language Project

**EXPANDED MASTER INPUT CHARACTER TRANSLATE TABLE**

<table>
<thead>
<tr>
<th>DEC</th>
<th>CODE</th>
<th>PRT(1)</th>
<th>PRT(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>,</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>b</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>z</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>Z</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>c</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>h</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>k</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>K</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>l</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>4</td>
<td>m</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>4</td>
<td>n</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>4</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>4</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>4</td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>4</td>
<td>x</td>
<td>/met</td>
</tr>
<tr>
<td>17</td>
<td>4</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>4</td>
<td>w</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>4</td>
<td>y</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>4</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>4</td>
<td>a</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>4</td>
<td>a</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>4</td>
<td>a</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>4</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>4</td>
<td>e</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>4</td>
<td>e</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>4</td>
<td>i</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>4</td>
<td>i</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>4</td>
<td>o</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>4</td>
<td>o</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>4</td>
<td>o</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>4</td>
<td>o</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>4</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>4</td>
<td>u</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>4</td>
<td>u</td>
<td></td>
</tr>
</tbody>
</table>

Characters 36-255 are not used and are coded 1, with PRT(1) and PRT(2) null.
SAMPLE INPUT
Input for the sample program consists of a data stream that is not keypunchable. However, it is similar in content to the sample given in the Description of Input section of the User Instructions.
SAMPLE OUTPUT

A
ALL
AND
BE
BETWEEN
CAUSE
CHARACTERS
DATA
DEFAULT
DELIMITERS
EXAMPLE
FOR
IGNORED
INPUT
IS
KGIC
MARKS
MOTT
OF
PARAGRAPH
PROGRAM
PROVIDED
PUNCTUATION
SAMPLE
TABLE
THE
THIS
TITLE
TO
VALUES
WILL

TYPES= 31
TOKENS= 43

KGIC: KEY GRAMEME IN CONTEXT LISTING
A TYPICAL KGIC RUN WITH ENGLISH DATA

A
DATA
TABLE
CHARACTERS
PARAGRAPH
ALL
VALUES
PROGRAM
SAMPLE
EXAMPLE
AND
PARAGRAPH
CHARACTERS
PARAGRAPH
MARKS
DATA
PUNCTUATION
DEFAULT
CAUSE

THERE ARE A TOTAL OF 23 OCCURRENCES OF GRAPHEME A
THERE ARE 19 UNIQUE OCCURRENCES OF GRAPHEME A

12
6/70
There are a total of 4 occurrences of grapheme B
There are 3 unique occurrences of grapheme B

There are a total of 6 occurrences of grapheme C
There are 6 unique occurrences of grapheme C

There are a total of 9 occurrences of grapheme D
There are 7 unique occurrences of grapheme D

There are a total of 22 occurrences of grapheme E
There are 18 unique occurrences of grapheme E

There are a total of 5 occurrences of grapheme F
There are 3 unique occurrences of grapheme F

There are a total of 5 occurrences of grapheme G
There are 4 unique occurrences of grapheme G
There are a total of 6 occurrences of grapheme H
There are 4 unique occurrences of grapheme H

A FEW STATISTICS

Total number of words processed (tokens) is 43
Number of different words (types) is 31
Type/token ratio is 0.721
Total number of characters processed is 204
Total number of unique occurrences of all characters is 162
Average length of word is 4.744

Frequencies of occurrence of each grapheme

<table>
<thead>
<tr>
<th>Grapheme</th>
<th>All Occurrences</th>
<th>Unique Occurrences</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Absolute</td>
<td>Relative</td>
</tr>
<tr>
<td></td>
<td>All Occurrences</td>
<td>Unique Occurrences</td>
</tr>
<tr>
<td></td>
<td>Absolute</td>
<td>Relative</td>
</tr>
</tbody>
</table>

A continuation of the frequencies of occurrence of each grapheme is shown below:

<table>
<thead>
<tr>
<th>Grapheme</th>
<th>All Occurrences</th>
<th>Unique Occurrences</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Absolute</td>
<td>Relative</td>
</tr>
<tr>
<td></td>
<td>All Occurrences</td>
<td>Unique Occurrences</td>
</tr>
<tr>
<td></td>
<td>Absolute</td>
<td>Relative</td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th>L</th>
<th>16</th>
<th>0.078</th>
<th>11</th>
<th>0.068</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>7</td>
<td>0.034</td>
<td>6</td>
<td>0.037</td>
</tr>
<tr>
<td>H</td>
<td>7</td>
<td>0.034</td>
<td>6</td>
<td>0.037</td>
</tr>
<tr>
<td>O</td>
<td>11</td>
<td>0.054</td>
<td>7</td>
<td>0.043</td>
</tr>
<tr>
<td>P</td>
<td>9</td>
<td>0.044</td>
<td>8</td>
<td>0.049</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>R</td>
<td>13</td>
<td>0.064</td>
<td>11</td>
<td>0.068</td>
</tr>
<tr>
<td>S</td>
<td>11</td>
<td>0.054</td>
<td>8</td>
<td>0.049</td>
</tr>
<tr>
<td>T</td>
<td>20</td>
<td>0.098</td>
<td>16</td>
<td>0.099</td>
</tr>
<tr>
<td>U</td>
<td>6</td>
<td>0.029</td>
<td>6</td>
<td>0.037</td>
</tr>
<tr>
<td>V</td>
<td>2</td>
<td>0.010</td>
<td>2</td>
<td>0.012</td>
</tr>
<tr>
<td>W</td>
<td>3</td>
<td>0.015</td>
<td>2</td>
<td>0.012</td>
</tr>
<tr>
<td>X</td>
<td>1</td>
<td>0.005</td>
<td>1</td>
<td>0.006</td>
</tr>
<tr>
<td>Y</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>Z</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
</tr>
</tbody>
</table>

continued
## A Typical KGIC Run with English Data

### Frequency of Occurrence of Each Grapheme

<table>
<thead>
<tr>
<th>Grapheme</th>
<th>2%</th>
<th>4%</th>
<th>6%</th>
<th>8%</th>
<th>10%</th>
<th>12%</th>
<th>14%</th>
<th>16%</th>
<th>18%</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>***</td>
<td>****</td>
<td>*****</td>
<td>******</td>
<td>*******</td>
<td>********</td>
<td>********</td>
<td>********</td>
<td>********</td>
<td>****</td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Grapheme Analysis

- The frequency of occurrence of each grapheme is indicated by the number of asterisks above each grapheme's name.
- The relative frequency is indicated by the percentage marks corresponding to the frequency levels.

### Summary

- The table provides a visual representation of the frequency of each grapheme, allowing for easy comparison and analysis.

---

**Notes:**

- The data is organized in a clear, tabular format, with each grapheme listed in a column.
- The relative frequency is shown using percentage marks, which help in identifying the commonality of each grapheme in the data set.
COST ESTIMATE
The computer costs for running the problem listed on the Sample Input were $3.30.

Charge to user = computer costs + network overhead
   = $3.30 + network overhead

CONTENTS—KGIC

pages
1–2 Identification & Abstract
3–10 User Instructions
11–16 I/O
17 Cost—Contents
FORMAC is a system for carrying out formal manipulations on mathematical expressions. This allows for the use of analytic as well as numeric techniques. The most important capability of FORMAC is its accommodation of mathematical expressions as symbolic entities at execution time. For example, the execution of the FORMAC program segment

```
LET( A = X + Y ** 2;
    B = 2;
    C = A/B + 2.8);
```

may be interpreted as assigning the alphanumeric value \((X+Y^2)/2 + 2.8\) to the FORMAC variable C.

FORMAC enables the user to analyze expressions by identifying coefficients, common denominators, lead operators, and some characteristics of the operands. Constants can be factored, left in a rational form, or converted to real notation. New expressions can be synthesized by the simplification, expansion and substitution of...
terms, as well as finding the derivatives of functions. The user can specify functions completely or partially in addition to making use of the PL/I functions. Procedures are available for transferring arguments between PL/I and FORMAC program segments.

REFERENCES
USER INSTRUCTIONS

The user should consult the reference listed.

FORMAC constants can be integer, floating-point, rational, or systems constants such as e, π, and i. Note that this latter constant enables FORMAC to handle imaginary expressions. FORMAC variables take on expressions as values that are retained as symbolic entities. Variables are either assigned or atomic—i.e., specified or unspecified. Expressions in FORMAC are written in essentially the same way as PL/I expressions but, whereas the latter evaluate to numbers, FORMAC expressions generally evaluate to new FORMAC expressions. There are four types of FORMAC functions: PL/I-like functions (such as SIN and COS), integer-valued functions (such as factorials and combinational expressions), user-defined functions, and function variables (unspecified functions). Functions with arguments that evaluate to constants can be either evaluated or retained as symbolic entities, at the user's option.

Program Deck

JOB-Control Language Statements and End-of-File Cards must begin in Col. 1; for FORMAC-PL/I statements, only Cols. 2-72 are used.

JOB Card will be punched by The Pennsylvania State University Computation Center personnel.

EXEC Card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>//</td>
</tr>
<tr>
<td>4-7</td>
<td>EXEC</td>
</tr>
<tr>
<td>9-21</td>
<td>LIBRARY,PROG=</td>
</tr>
<tr>
<td>22-27</td>
<td>FORMAC</td>
</tr>
</tbody>
</table>

Data Card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-17</td>
<td>//DATA.INPUT DD *</td>
</tr>
</tbody>
</table>

Program Deck

JOB Card
EXEC Card
Data Card

...
REFERENCES

SAMPLE INPUT

//ANT12145.c JOS oc4nn2,Q=hnftam0.C21..1KLF Ho...--
// EXL LIBRARY.UNITDEFINITION
//DATA INPUT ON

IMPORT PARENTHESES

COMMENT IN THE FOLLOWING EXAMPLES "STRING" INDICATES THE INPUT.

COMMENTS: THE UNIMPLIRED 2-DIMENSIONAL STRING INPUT IS THE

COMMENTS RESULT OF PROCESSING BY THE INTERPRETATIVE FORTRAN SYSTEM.

/*

COMMENT GENERAL EXAMPLE:

IR = A*SINIA+11 + .AF-44,111G(fnclY)),

SPAN:

/*

COMMENT FUNCTION DEFINITION AND EVALUATION,

Y = X**2, X = 1/30

SPAN:

/*

COMMENT FUNCTION DEFINITION AND EVALUATION,

Y = (-1)**x + 2*SINItI21/'

SPAN:

/*

COMMENT FUNCTION DEFINITION AND EVALUATION,

SOURCE: MAC,

SPAN:

/*

COMMENT FUNCTION DEFINITION AND EVALUATION,

SOURCE: MAC,

SPAN:

/*

COMMENT FUNCTION DEFINITION AND EVALUATION,

SOURCE: MAC,

Spain:

/*

COMMENT FUNCTION DEFINITION AND EVALUATION,

SOURCE: MAC,

Spain:

/*

COMMENT FUNCTION DEFINITION AND EVALUATION,

SOURCE: MAC,

Spain:

/*

COMMENT FUNCTION DEFINITION AND EVALUATION,

SOURCE: MAC,

Spain:
STRING=*"SET (LINELENGTH=SHORT)"*
LINELENGTH= 64

IN THE FOLLOWING EXAMPLES "STRING=" INDICATES THE INPUT STRING; THE UNDERLINED TWO-DIMENSIONAL EDITED OUTPUT IS THE RESULT OF PROCESSING BY THE INTERPRETATIVE FORMAC SYSTEM.
GENERAL EXAMPLE.

STRING=IA = $(X^2 + 7/3)^2$
A = $8P \left( X + 7/3 \right)$

STRING=IB = 3*SIN(A+1) + .3E-4*LOG(COS(Y))
B = .3E-04 LN ( COS ( Y ) ) + 3 SIN ( 8P ( X + 7/3 ) + 1 )

UNSPECIFIED FUNCTIONS AND DERIVATIVES.

STRING=IA = F.3*X**2 + LOG(X)**2
A = F.1 ( 3* X^2 + LN ( X ) )

STRING=IB = DERIVA(A,X,2)
B = 3*F .( 3* X ) X - 1 / X + 6 F .( 3* X )

STRING=IC = DERIVA(ATANH(X**2) + X,1)
C = 2 X / ( - X + 1 ) + 1

STRING=ID = ORV( U, V, W, 43131, 2121, 321211)

TRANS OPTION FOR TRANSCENDENTAL FUNCTION EVALUATION.

STRING=IPSET(INOTRANS)
OPTSET(TRANS)
STRING=IA = 1/31
A = 1/3

STRING=IB = SIN(X**4) + LOG(2*A) + SIN(8P/6)
B = LN ( 2/P ) + SIN ( 1/6 #P ) + SIN ( X + 1 )

STRING=IPSET(TRANS)
OPTSET(TRANS)
STRING=IC = B**4
C = SIN ( X + L ) + 0.0453489

INT OPTION FOR COMBINATIONS, FACTORIAL, AND EVAL CAPABILITY.

STRING=IPSET(NOINT)
OPTSET(INT)
STRING=IV = COMB(N,K-1) + CUMB(N,K)
Y = ( N! K ) + ( N! K - 1 )

continued
FUNCTION DEFINITION AND EXPANSION APPLYING MULT. AND DIST. LAWS

STRING="A = EVAL(Y, X, 5, X, 3)";
A = (51, 2 1 + (51, 3)

STRING="OPTSETINT1":
OPTSETINT1

STRING="B = EVAL(Y, N, 5, X, 3)";
B = 20

STRING="FNCIG) = 6(1) + 2*SIN(6 (2) )^2:
G = 2 SIN (6 (2) ) + 6(1)

STRING="GOTX**2 + 1, Z)";
2
P = X + 2 SIN (Z) + 1

STRING="A = (X+4)**2 + 3*(2+2*Z) + (Z-3)";
A = 3 (Z - 3 ) (Z + 2 X) + (X + 4)

STRING="B = EXPAND(A)";
2
B = - 9 Z - 10 X + 6 X Z + 1 Z + X + 16

COMMON DENUM, NUM, DENOM, LOW AND HIGH POWER AND COEFF

STRING="A = X/Z + Z + X/W";
A = Z / X / W * X / Z

STRING="B = CODE(A)";
B = (X W + (X + Z W) Z) / (Z W)

STRING="Y = (Z*X**5 + 7/3*X**31/SIN(X) + COS(X))";
Y = (X Z + 7/3 X ) / (SIN (X) + COS (X) )

STRING="A=NUM(Y)";
A = X Z + 7/3 X

STRING="B = DENOM(Y)";
B = SIN (X ) + COS (X )

STRING="C = HIGHPOW(A, X)";
C = 5

STRING="D = LOWPOW(A, X)";
D = 3

continued
STRING='E = COEFF(A,x**9); F = F
-------

FOIT FEATUIF.
STRING='E = C1 + C2*X + C3*X**2; A = C1 + X C2 + X C3
-----------

STRING='DPESTWNIEEIT11'; DPESTWNIEEIT11
STRING='C = C1 + C2X + C3X**2
 -----------

INTEGER AND RATIONAL CONSTANTS MAY BE DEVELOPED OR SPECIFIED
WITH AS MANY AS 2295 SIGNIFICANT DIGITS.
STRING='Y = 1167213468759/4 * x**174; T = X174; IX = 23345642693755
-----------

STRING='A = DERIVITY,x,51';
A = X[X0 = 3513749301039751193720

STRING='R = EVALIA,x,241/121';
B = (1207429010674375174601225257162239489764211068979817372
-----------

5076451194821708423266313512411702559192071390186446167420740
786691946246127473136951547911471720394729536408990411693012
--------------
429176574589932138974346070415629/5973741656292027805246157
--------------
20574212039946114092131508139..029149231306937607988059
212598441724019926425784032423411242618443039327804411302464
--------------
1094081616411761617257109361

Y=0: X=-1, X=1; Y=1: -1.0(X<X); "=COS(X); D=K4.
STRING='Y = STEP(-1,X,0) + STEP(0,X,11) * COS(X); Y = COS(X)*STEP(0,X,11) + STEP(-1,X,C)
--------------

*NORMAL STOP WITH INPUT TO FMACUT EXHAUSTED**

HASP VERSION 2.42C 51 CARDS READ 179 LINES PRINTED 0 CARDS PUNCHED 5 SECS NET CPU TIME
COST ESTIMATE

For the job listed on the Sample Output, the total running time on the central-processor unit was 10 seconds, at $0.11 per second. The chargeable computer time was $1.10.

Charge to user = computer time + network overhead
              = $1.10 + network overhead
FUNCTIONAL ABSTRACT

The FORMAC version available at The Pennsylvania State University is an interpretative system; i.e., FORMAC expressions are treated as character strings that are evaluated and executed at run time only. These character strings may be specified by reference to a PL/1 character-string variable. The PL/1 variable, in turn, may be constructed by execution of the PL/1 program in which the FORMAC statements are embedded. FMACUT merely capitalizes on the last-named feature. FORMAC statements are read by the preprocessed, precompiled, and prelink-edited program and are passed to the FORMAC package as character-string arguments. Note that this is made possible also because FORMAC variables need not (in fact cannot) be declared in the PL/1 program. Reference 1 should be consulted concerning more-detailed PL/1 FORMAC information.

FMACUT will execute FORMAC statements that are supplied as input data. It effectively divorces FORMAC from PL/1 insofar as the user is concerned. This has two advantages: (1) the user need have no knowledge of PL/1 to use FORMAC and (2) FORMAC is available without preprocessing, compilation, or link-editing (which continued
saves approximately 50 seconds of run time for a null job on the IBM 360/67). The corresponding disadvantage is that PL/1 facilities and those (few) FORMAC statements involving a PL/1 direct interface are not available to the user.

REFERENCES
Tobey, R., Baker, J., Crews, R., Marks, P., and Victor, K., "PL/1 FORMAC Interpreter," IBM Publ. 360D 03.3.004 (1967).
USER INSTRUCTIONS

FMACUT may be executed via the catalogued procedure LIBRARY. The input is freeform; i.e., card boundaries are ignored. FORMAC statements are written within single quotes ('') rather than within balanced parentheses following the word LET. For example, instead of using LET(X=A+B* COS(PL/2)), use 'X=A+B*COS(PL/2)'. Only one FORMAC statement may be enclosed within a given pair of single quotes. For example, instead of using LET(X=A**2+C*A-D; Y=COEFF (X,A**2)), use 'S=A**2+c*a-D' 'Y=COEFF(X,A**2)'. NOTE: The single quote terminating one FORMAC statement by one or more blanks and/or one comma. An individual FORMAC statement, exclusive of its enclosing quotes may be no longer than 800 characters in length, including blanks.

To facilitate setting FORMAC options, the OPTSET command may be used. OPTSET followed by the desired options (optionally enclosed within parentheses) must be enclosed within single quotes. For example, 'OPTSET INT EXPAND' and 'OPTSET(PROPER)' are acceptable. All options, including PRINT and NOPRINT, may be used and may be set, changed, and reset as often as desired. The default options are LINELENGTH=LONG (i.e., 120), PRINT, TRANS, INT, NOEXPND, EDIT, and IMPROPER. OPTSET is a reserved keyword.

To provide the capability of adding comments to the output, the command 'COMMENT-any comment' may be used. For example, 'COMMENT- PROOF BY INDUCTION' would result in this comment being printed. COMMENT is a reserved keyword and, like OPTSET, may not be used as a variable name.

To adjust the number of characters that will be printed out per line, two options are available: 'OPTSET(LINELENGTH=LONG)' will use a full page (120 characters per line) and 'OPTSET(LINELENGTH=SHORT)' will print out in a form suitable for 8½ X 11 in. paper (72 characters per line). When using the latter option, comments should be limited to 72 characters, including COMMENT.

Capabilities and Limitations

The commands SAVE and ATOMIZE are not available. In general, only those FORMAC statements that would normally be enclosed within parentheses following LET may be used. In particular, the features ARITH, INTEGER, IDENT, EXCH, and double quotes (")) involving a direct PL/1 to FORMAC interface are not available. OPTSET and COMMENT are available as explained above. If a FORMAC expression is incorrectly structured with respect to syntax or logic, as described in the FORMAC manual, the statement is bypassed, an error message is issued, and processing continues. One special character used by FORMAC—namely, the exclamationation point (!)—is
not available on the QN (normal) print train. Therefore, this character, used as a factorial sign when the INT option is off, will print as a blank. This limitation, however, is not problem when using a terminal via RJE.

The transcendental-function names that are available are the following: SIN, COS, SINH, COSH, ATAN, ATANH, ATAN(x,y), LOG, LOG10, LOG2, ERF, EXP, SQRT, SIND, COSD, TAN, TAND, TANH, ATAND, ATAND(x,y), and ERFC. In addition, as part of the functional expression, the constants #P=Pi, #E=e (base of natural logs), and #i=√-1 will be recognized. [E.g., FORMAC will transform SIN(#P/6) before computation to 1/2.] The functions FAC(n) = n! and COMB(n,n1,n2,...,nk) = [n(n-1)⋯(n-n1−⋯−nk+1)]/[n!n1!⋯nk!] also are available for n and k non-negative integers. The absolute-value function ABS(X) is not available; however, this operation may be effected by SQRT(X**2).

The number of significant digits in any floating-point constants may not exceed eight. The number of variables and degree of expansion are limited only by storage. Most of these limitations can be eliminated for a particular case by modifying FMACUT.

Program deck and Control Cards are the same for FMACUT and FORMAC, except that Cols. 22–27 of the EXEC Card contain FMACUT.

REFERENCES
1. Tobey, R., Baker, J., Crews, R., Marks, P., and Victor, K., "PL/1 FORMAC Interpreter," IBM Publ. 360D 03.3.004 (1967).
SAMPLE INPUT

// EXEC LIBRARY9PROG=FMACUT
//DATA.INPUT DO *

A = (X**2 + 7/3)*X
B = 3*SIGN(A+1) + .3E-4*LOG(COS(Y))
C = DERIV(A*X**2)
D = DERIV(B,X**2) + X*X
E = TRANS OPTION FOR TRANCENDENTAL FUNCTION EVALUATION

OPTSET(INOTRANS)
A = 1/3
B = SIN(X**4) + LOG(2*A) + SIN(P/6)
OPTSET(INTRANS)
C = B

OPTSET(ININT)
Y = COMB(N,K-1) = COMB(N,K)
A = EVALY(N,K,3)
OPTSET(INTRANS)
B = EVALY(N,K,3)

OPTSET(INTRANS)

Y = COMB(N,K-1) + COMB(N,K)
A = EVALY(N,K,3)

OPTSET(INTRANS)

Y = f(x)
A = f(x)

OPTSET(INTRANS)

Y = g(x)
A = g(x)

OPTSET(INTRANS)

Y = h(x)
A = h(x)

OPTSET(INTRANS)

Y = i(x)
A = i(x)

OPTSET(INTRANS)

Y = j(x)
A = j(x)

OPTSET(INTRANS)

Y = k(x)
A = k(x)

OPTSET(INTRANS)

Y = l(x)
A = l(x)

OPTSET(INTRANS)

Y = m(x)
A = m(x)

OPTSET(INTRANS)

Y = n(x)
A = n(x)

OPTSET(INTRANS)

Y = o(x)
A = o(x)

OPTSET(INTRANS)

Y = p(x)
A = p(x)

OPTSET(INTRANS)

Y = q(x)
A = q(x)

OPTSET(INTRANS)

Y = r(x)
A = r(x)

OPTSET(INTRANS)

Y = s(x)
A = s(x)

OPTSET(INTRANS)

Y = t(x)
A = t(x)

OPTSET(INTRANS)

Y = u(x)
A = u(x)

OPTSET(INTRANS)

Y = v(x)
A = v(x)

OPTSET(INTRANS)

Y = w(x)
A = w(x)

OPTSET(INTRANS)

Y = x(x)
A = x(x)

OPTSET(INTRANS)

Y = y(x)
A = y(x)

OPTSET(INTRANS)

Y = z(x)
A = z(x)
SAMPLE OUTPUT

STRING*OPTSET I LINELENGTH= SHORT) ;
LINFL num= .4

IN THE FOLLOWING EXAMPLES "STRING=" INDICATES THE INPUT
STRINGS THE UNDERLINED TWO-DIMENSIONAL OUTPUT IS THE
RESULT OF PROCESSING BY THE INTERPRETATIVE FORMAC SYSTEM.

GENERAL EXAMPLE.
STRING*A = (X**2 + 7/3)**P ;
2
A = X**P ( X + 7/3 )

-----------------

STRING*R = 3*5IN(A+1) + .3E-4*LOG(COS(Y+1)) ;
2
R = 3*5*0.4 LN ( COS ( X ) ) + 9 SIN ( Y**P ( X + 7/3 ) )

-----------------

UNSPECIFIED FUNCTIONS AND DERIVATIVES.
STRING*A = F.3*(X**2) + LOG(X) ;
F = F. 3 X**2 + LN ( X )

-----------------

STRING*B = DERIV(A,X,2) ;
F 2 2
B = 36 F 1 3 X**2 + 1 /X + 6 F .1 3 X

-----------------

STRING*C = DERIV(TANH(X**2) + X**X) ;
X
C = 2 X / ( -X + 1 ) + 1

-----------------

STRING*D = DERIV(U,V,W,X1,X2,X3) ;
U2 V2 W2
D = H ( 1 2 3 ) .1 U V W

-----------------

TRANS OPTION FOR TRANSCENDENTAL FUNCTION EVALUATION.
STRING*OPTSET(NOTRANS) ;
OPTSET(N1TRANS)
STRING*A = 1/3*;
A = 1/3

-----------------

STRING*B = SIN(X**4) + LOG(2*A) + SIN(0/6) ;
4
B = LN ( 2/3 ) + SIN ( 1/6 #P ) + SIN ( X )

-----------------

STRING*OPTSET(TRANS) ;
OPTSET(TRANS)
STRING*C = B ;

continued
\[ C = \sin (4 \times x) + 0.0451449 \]

---

**INTERNAL OPTIONS FOR COMBINATIONS, FACTORIAL, AND EVAL CAPABILITY.**

**STRING** A = !PTSET(1INT1);  
**OPTSET** A = !INT1;

\[ Y = \binom{n}{k} + \binom{n}{k-1}; \]

**STRING** A = EVAL(Y, N, S, K, 3);  
**OPTSET** A = (!INT1);

\[ A = \binom{n}{k} + \binom{n}{k-1}; \]

**STRING** A = !PTSET(1INT1);  
**OPTSET** A = !INT1;

\[ Y = \binom{n}{k} + \binom{n}{k-1}; \]

**STRING** A = EVAL(Y, N, S, K, 3);  
**OPTSET** A = (!INT1);

\[ A = \binom{n}{k} + \binom{n}{k-1}; \]

**FUNCTION DEFINITION AND EXPANSION APPLYING MULT. AND DIST. LAWS**

**STRING** FNC(G) = 2\sin(4(2)) + 2;  
\[ G = 2\sin (4(2)) + 4(1); \]

**STRING** B = 2 \times 2;  
\[ B = 2 \times 2\sin (4(2)) + 1; \]

**STRING** C = (4+4)(2) + 3(2)(4+2);  
\[ C = (4+4)(2) + 3(2)(4+2); \]

**STRING** D = X + 2 \times 2\sin (4(2)) + 2;  
\[ D = X + 2 \times 2\sin (4(2)) + 2; \]

**COMMON DENOM., NUM., DENOM., LOW AND HIGH POWER, AND COEFF**

**STRING** A = X/Z;  
\[ A = X/Z; \]

**STRING** B = CODEM(11);  
\[ B = (X + (X + Z \times Z)) / (Z \times Z); \]

---

continued...
STRING=STRING=
Y = (2*X**5 + 7/3*X**31/ISIN (X) + COS(X))
5
Y = 1 X 2 + 7/3 X 1 / (SIN (X) + COS (X))

STRING=STRING=
A = X 2 + 7/3 X
5

STRING=STRING=
A = DENOM (Y)
R = SIN (X) + COS (X)

STRING=STRING=
C = 5

STRING=STRING=
D = LOWP (A, X)
D = 3

STRING=STRING=
E = LOWP (A, X)
E = 2

EDIT FEATURE.

STRING=STRING=
A = C1 + C2 X + C3 X**2
A = C1 + X C2 + X C3

STRING=STRING=
OPTSET (NOEDIT)
OPTSET (NOEDIT)

STRING=STRING=
A = A
A = C1 + C2 X + C3 X**2

INTEGER AND RATIONAL CONSTANTS MAY BE DEVELOPED OR SPECIFIED
WITH AS MANY AS 2995 SIGNIFICANT DIGITS.

STRING=STRING=
Y = 11617713468795/5 * X**174
Y = X**174

STRING=STRING=
A = DERIV (Y, X, 5)
A = X**169

continued
STRING= \( B = \text{EVAL}(A \times x, 2^{11/12}) \);  
\( R = (720752901067547517480962225271612235848764211968938173777) \);  
\( 03941030005920105250323273475367893232233295955704333346085570 \);  
\( 5074651984217084232060915657631702550192071908186546353420740 \);  
\( 7386691852666127473136954558711457201988872535864890611693012 \);  
\( 42912216756489312388943656704111259573741656293202778092651157 \);  
\( 2007421252034930611409217151358334517052910452310306937607988039 \);  
\( 72580644172401592984657884242342124426413443035327084411307454 \);  
\( 10940816116176161725710336 \);  
\( Y = 0: x < -1 \);  
\( Y = 1: -1 < x < 1 \);  
\( Y = \text{COS}(x) \);  
\( \text{STRING} = Y = \text{STEP}(-1, x, 0) + \text{STEP}(0, x, 1) \times \text{COS}(x) \);  
\( Y = \text{COS}(x) \times \text{STEP}(0, x, 1) + \text{STEP}(-1, x, 0) \);  

**NORMAL STOP WITH INPUT TO FNACUT EXHAUSTED**

HASP VERSION 2.02  
52 CARDS READ  
216 LINES PRINTED  
0 CARDS PUNCHED  
5 SECS NET CPU TIME
COST ESTIMATE
For the job listed on the Sample Input, the total running time on the central-processor unit was 10 seconds, at $0.11 per second. The chargeable computer time was $1.10.

Charge to user = computer time + network overhead
              = $1.10 + network overhead

CONTENTS—FMACUT
pages
1-2 Identification & Abstract
3-4 User Instructions
5-9 I/O
11 Cost—Contents
DESCRIPTIVE TITLE  Collection of Statistical Routines

CALLING NAME  STPAC

INSTALLATION NAME  The Pennsylvania State University Computation Center

AUTHOR(S) AND AFFILIATION(S)  William H. Verity
Nancy C. Daubert
The Pennsylvania State University Computation Center

LANGUAGE  FORTRAN IV

COMPUTER  IBM System 360/67

PROGRAM AVAILABILITY  Decks and listings presently available

CONTACT  William H. Verity, 104 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

FUNCTIONAL ABSTRACT

STPAC consists logically of two sections: a monitor and a collection of statistical routines. The monitor is a program that will read all Control Cards, save the original data if required or requested, save any intermediate results that may be needed later on, and then transfer control to the first routine requested. When this routine is finished, it returns control to the monitor, which then looks to see if there is more work to be done. This work can be of several different types, as follows.

1. Using the same data deck (but possibly different columns of it), execute the same routine again.

2. Using the same data deck, execute another routine. Thus, one deck can be subjected to several types of analyses during one run.

3. Execute another program, possibly the same one as before, on a new set of data.

4. Using the results of a previous routine, continue to another program that takes these results as input; e.g., output from correlation can be the input to regression or factor analysis.

The statistical routines available include a variety for summary continued
of data, frequency analysis, tests of significance, analyses of variance, and rank and product-moment correlation coefficients. See individual abstracts for EIN Nos. 000 0051(a)–(s) for the details of the individual statistical routines.

REFERENCES

See individual abstracts for EIN numbers 000 0051 (a)–(s).

STPAC ROUTINES

<table>
<thead>
<tr>
<th>EIN No.</th>
<th>Calling Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>000 0051(a)</td>
<td>FAWCS</td>
</tr>
<tr>
<td>(b)</td>
<td>STSUM</td>
</tr>
<tr>
<td>(c)</td>
<td>TTEST</td>
</tr>
<tr>
<td>(d)</td>
<td>PPMCR</td>
</tr>
<tr>
<td>(e)</td>
<td>SIGPP</td>
</tr>
<tr>
<td>(f)</td>
<td>PARCOR</td>
</tr>
<tr>
<td>(g)</td>
<td>CANON</td>
</tr>
<tr>
<td>(h)</td>
<td>UPRG/DRREG</td>
</tr>
<tr>
<td>(i)</td>
<td>FANAL</td>
</tr>
<tr>
<td>(j)</td>
<td>VARMX</td>
</tr>
<tr>
<td>(k)</td>
<td>PHICO</td>
</tr>
<tr>
<td>(l)</td>
<td>KETAU</td>
</tr>
<tr>
<td>(m)</td>
<td>SPRI0</td>
</tr>
<tr>
<td>(n)</td>
<td>MNWHT</td>
</tr>
<tr>
<td>(o)</td>
<td>KRWAL</td>
</tr>
<tr>
<td>(p)</td>
<td>ANOVUM</td>
</tr>
<tr>
<td>(q)</td>
<td>ANOVES</td>
</tr>
<tr>
<td>(r)</td>
<td>AOVRM</td>
</tr>
<tr>
<td>(s)</td>
<td>BARTL</td>
</tr>
</tbody>
</table>
USER INSTRUCTIONS

To use any of the programs of the statistical package, *STPAC MUST BE SPECIFIED AS THE LIBRARY NAME*. Programs available as part of this package are identified and described in separate writeups [EIN Nos. 000 0051(a)-(s)].

Control Cards and Their Functions

Problem information is given through the use of Control Cards. Each Control Card transmits one type of information and conveys to the user its function. The information on a Control Card is contained in one or more fields, where a field is defined to be one or more contiguous nonblank characters. Fields are separated from one another by one or more blank spaces or commas, thus allowing the user to punch the Control Cards without regard to specific columns, although the fields must be in the specified order. The only restriction is that the first field must begin in Col. 1. If any field contains special characters or blanks, it must be enclosed in single quotation marks.

To execute any given analysis, two types of Control Cards will be needed: those applicable to all of the programs in the package and those specific to the program at hand. The Control Cards that can apply to *all* programs are described next. The Control Cards that are peculiar to an *individual* program are described in the appropriate program writeup. Information printed in capital letters must appear on the Control Card letter for letter as it is typed and all fields must be in the order indicated.

Execute Card (required for each problem)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–7</td>
<td>EXECUTE</td>
</tr>
<tr>
<td>9–13</td>
<td>Name of program to be executed</td>
</tr>
</tbody>
</table>

*NOTE:* This must be the first card for each problem under STPAC.

Comments Card(s) (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>*</td>
</tr>
<tr>
<td>2–80</td>
<td>Information to be printed on a page immediately preceding output (for documentation)</td>
</tr>
</tbody>
</table>

Any number (or none) of these cards may be used.

*continued*
Title Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>TITLE</td>
</tr>
<tr>
<td>6-80</td>
<td>Heading to be printed on each section of output</td>
</tr>
</tbody>
</table>

Variable Card (may or may not be required)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-9</td>
<td>VARIABLES</td>
</tr>
<tr>
<td>11-12</td>
<td>Number of variables for the problem</td>
</tr>
<tr>
<td>14-22</td>
<td>SEQUENCED</td>
</tr>
<tr>
<td>24-25</td>
<td>Number of variables on each Data Card (&lt; number in Cols. 11-12)</td>
</tr>
<tr>
<td>27-32</td>
<td>PERCARD</td>
</tr>
</tbody>
</table>

NOTE: If Cols. 14-32 of this card are specified, then it is inferred that case and sequence numbers (strictly increasing only) are on each Data Card, each case number being identified in the first card field and the sequence number within the case being given in the second card field. If the case and sequence numbers are missing (or it is not desired that they be checked), entries in these columns must be omitted. In this instance, either all variables fit on one card or the Format Card (see below) controls the reading of more than one card.

Name Card(s) (optional; first form)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-80</td>
<td>VARIABLE NAMES i=xxx,j=yyy, etc.</td>
</tr>
<tr>
<td></td>
<td>or</td>
</tr>
<tr>
<td></td>
<td>TREATMENT NAMES i=xxx,j=yyy, etc.</td>
</tr>
</tbody>
</table>

One or more of these cards may be used to assign alphanumeric names to each variable and/or treatment. Variable (treatment) i will be called xxx, j will be called yyy, etc., where xxx and yyy represent some combinations of eight or fewer characters, at least one of which must be nonnumeral. If the card is omitted, variables (treatments) will be numbered sequentially. It is also possible to name some variables (treatments) and to allow the remaining ones to be numbered. In addition, xxx, yyy, etc., may be labelled with a two-word term if it totals no more than eight characters and has single quotation marks around it.

continued
Examples (starting in Col. 1)

VARIABLE NAMES 1=AGE, 2=PARTY
TREATMENT NAMES 1=HEIGHT, 2=HAIR COL'

Name Card(s) (optional; second form)

Columns    Contents
1-80       VARIABLE NAMES TRANSFER
or
TREATMENT NAMES TRANSFER

This card indicates that the names from the immediately preceding problem are to be used in the present problem (especially useful when results from one problem are data for another).

Format Card(s) (may or may not be required)

Columns    Contents
1- 6       FORMAT
8          Number of cards used in format specification (<3)
10-80      Beginning of format specification (any valid FORTRAN FORMAT)

NOTE: If the number of cards required is one, the format specification may begin in Col. 8 (and the 1 need not be specified). If more than one card is required, the word FORMAT need not be repunched, thus making all 80 columns of the second (and third) card available for use.

Begin-Data Card (required if Data Cards are to be used; must be omitted if data from previous problem are to be used)

Columns    Contents
1-10       BEGIN DATA

This card must immediately precede the data deck. Previous data are no longer available after this card has been read.

End-Data Card (required if Data Cards are being used)

Columns    Contents
1- 8       END DATA

This card must immediately follow the data deck.
Same-Data Card (optional)

Columns    Contents
1-9   SAME DATA

This card indicates that the last set of raw data is to be used, allowing a data deck to be used over and over again. When this card is used, there cannot be either a Begin- or End-Data Card.

Input Card (optional)

Columns    Contents
1-10   INPUT FROM
12-16 Name of program already run (during the present run) from which the input to this program is coming (used when the results of one program are the input to another). There may be cases in which the input may be from both another program and card input. Then, both the INPUT FROM and BEGIN DATA–END DATA sequence would be used.

Stop Card (required once for each submission to the package)

Columns    Contents
1-4   STOP (indicating the end of input to STPAC)

Execution of all programs will terminate after this card is read.

Job Deck Structure

EXECUTE xxxxx (required)
Comments Card(s)
Title Card
Variable Card
Name Card(s)
Format Card(s)
Other cards (required or optional for program xxxxx)
Same-Data Card
Input Card
Begin-Data Card
Data Card(s)
End-Data Card
Stop Card (required at end of STPAC input)

May be in any order

Repeat for as many problems (programs) as necessary

NOTE: For each problem, there must be either a Same-Data Card, an Input Card, or a deck of Data Cards (beginning with the Begin-Data Card and ending with the End-Data Card).
SAMPLE INPUT/OUTPUT

Sample Input and Output are specific to the various routines. See items 000 0051(a)-(s).
COST ESTIMATE
Inappropriate for STPAC alone; see individual routines EIN Nos. 000 0051 (a)-(s)

CONTENTS—STPAC

pages
1-2 Identification & Abstract
3-7 User Instructions
7 I/O
9 Cost—Contents
DESRIPTIVE TITLE         Frequency Analysis with Chi Square

CALLING NAME             (STPAC) FAWCS

INSTALLATION NAME         The Pennsylvania State University Computation Center

AUTHOR(S) AND AFFILIATION(S)

C. B. Broderick
Department of Family Relations, The Pennsylvania State University

William H. Verity
The Pennsylvania State University Computation Center

LANGUAGE                  FORTRAN IV

COMPUTER                  IBM System 360/67

PROGRAM AVAILABILITY      Decks and listings presently available

CONTACT                   William H. Verity, 104 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
                          Tel.: (814) 865-9527

FUNCTIONAL ABSTRACT

FAWCS computes (1) discrete, two-dimensional frequency-distribution tables (often called contingency tables) whose cells contain an actual frequency and, optionally, (2) expected frequencies and/or percentages. There is no missing-data code inherent to the program, but the user may specify that certain categories of his variables be ignored or combined with one another. If the user has any 2x2 tables, he may request that the Yates correction factor be used in computing his chi-square value.

FAWCS will crosstabulate one variable, the row variable, with variables called column variables. It can work with up to 50 total variables for any one problem, separately crosstabulating the row variable with each of the column variables. The row variable can have up to 31 categories where 0 is a valid category. Thus, the row variable can be coded 0 through 30. Each column variable can have a maximum of 13 categories where, again, 0 is a valid code. Therefore, column variables can be coded 0 through 12. Any data outside of these ranges will be ignored.

continued
REFERENCES


USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

In addition to the Control Cards applicable to all programs in STPAC, FAWCS uses the Control Cards that are described in the following. There is no standard format for raw data, but if totals data are used they must be punched as described under the PUNCH option. A Variables Card is required for FAWCS.

Row-Variable Card (required)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-12</td>
<td>ROW VARIABLE</td>
</tr>
<tr>
<td>14-80</td>
<td>Variable to be used as row variable. All others will become column variables. It may be specified by either the name or the number assigned to it. (If Name Cards were not used, the reference to the variable by name would be permitted.)</td>
</tr>
</tbody>
</table>

Percentages Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-11</td>
<td>PERCENTAGES</td>
</tr>
<tr>
<td>13-15</td>
<td>ROW or one of these</td>
</tr>
<tr>
<td>13-18</td>
<td>COLUMN</td>
</tr>
</tbody>
</table>

This card indicates that percentages are to be computed on the basis of row (column) totals. Only one choice may be made for any one problem.

Chi-Square Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-10</td>
<td>CHI-SQUARE</td>
</tr>
</tbody>
</table>

This card indicates that a chi-square value is to be printed for each table, as well as the probability of exceeding this value by chance.

continued
Yates-Correction-Factor Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>YATES</td>
</tr>
</tbody>
</table>

This card indicates that the Yates correction factor should be used in any 2x2 tables that are developed. It will be used only in computing the chi square.

Expected-Frequency Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-8</td>
<td>EXPECTED</td>
</tr>
</tbody>
</table>

This card causes the expected cell frequencies to be printed.

Punch Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>PUNCH</td>
</tr>
</tbody>
</table>

This card causes the raw cell frequencies to be punched onto cards so that it is possible to go back and either combine or ignore different or new combinations of row and column categories without re-reading the raw data. In order that a minimum number of cards have to be repunched by hand, the following cards are punched.

EXECUTE FAWCS
VARIABLES 2
VARIABLE NAMES 1=xxxxx,2=wwwww
ROW VARIABLE xxxxx
TOTALS (see below)
BEGIN DATA
: actual totals from the raw table
END DATA

Thus, all that must be added is a Title Card. After adding or ignoring any other cards (including optional Control Cards), the deck is ready for resubmission. Totals are punched according to a format of 2014, as follows.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>Row number</td>
</tr>
<tr>
<td>6-9</td>
<td>Column number</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
</tr>
<tr>
<td>14-17</td>
<td>Total for this row-column combination</td>
</tr>
</tbody>
</table>

continued
NOTE: The PUNCH option is recommended only for exceptionally large amounts of data. Otherwise, it is easier simply to resubmit the raw data.

Totals Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-6</td>
<td>TOTALS</td>
</tr>
</tbody>
</table>

This card indicates that totals punched in a previous run of FAMCS (or otherwise) are to be used. Totals must be punched in the 2014 format described above.

Ignore-Categories Card (optional; one or more as required—first form)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-80</td>
<td>IGNORE xxxxx CATEGORIES i j k</td>
</tr>
</tbody>
</table>

This card causes categories i,j,k,... of the variable xxxxx to be ignored, where xxxxx is either the name or the number of a variable. A preliminary table with all categories will be printed, followed by a final table in which the specified categories will not appear. All chi squares, percentages, and expected frequencies will be based on the final table.

Ignore-Categories Card (optional; second form)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-80</td>
<td>IGNORE $COLUMN$ CATEGORIES i j k</td>
</tr>
</tbody>
</table>

This card indicates that categories i,j,k,... of all column variables are to be ignored. To ignore row-variable categories, the first form of this card must be used.

Combine-Categories Card (optional; one or more as required—first form)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-80</td>
<td>COMBINE xxxxx CATEGORIES i j k</td>
</tr>
</tbody>
</table>

This card causes categories i,j,k,... of variable xxxxx to be combined to form a new category, where xxxxx is xxxxx is either the name or the number of a variable. The new category will carry a number that is the smallest of those combined.

continued
Combine-Categories Card (optional; second form)

Columns   Contents
1-80       COMBINE $COLUMN$ CATEGORIES i j k ...

This form indicates that, for each column variable, categories i,j,k,... are to be combined to form a new category. Again, to combine row-variable categories, the first form of this card must be used.

NOTE: If there is an inconsistency in the Combine- and Ignore-Categories Cards, only the preliminary table will be printed and cards will be punched.

Format Card (required for raw data; not permitted for totals)

The form of this card is as described in the general discussion of Control Cards for STPAC. For FAWCS, the format can specify only integer data (I notation). If case and sequence numbers are required, the format must allow for them.

Description of Output

The printed output for a FAWCS problem is a series of two-dimensional tables, one table for each column variable. Each table is headed by the title that the user has supplied. The categories of each column variable are listed across the top of the table, along with the name of the column variable. The categories of the row variable are listed down the lefthand margin of the table, along with the name of the row variable. The body of the table contains the observed frequencies for each cell and percentages and expected frequencies if requested. Below each table, if requested, the chi-square value is printed, along with its degrees of freedom and the probability of exceeding this value by chance.

If any Combine- or Ignore-Categories Cards were present, a preliminary table is printed. It contains the same title and headings, but only the observed frequencies appear in the body of the table. At the bottom of this table are listed the categories that are to be ignored and combined. Then, a final table in which the combining and ignoring has taken place is printed on a new page. All calculations are based on this final table.

Card output is produced where applicable, as described under the Punch Card.
EDUCATIONAL INFORMATION NETWORK

EDUCOM

000 0051 (a)

40.

SAMPLE INPUT
ul
CD
CD
CD
CD
CD

F-A

liA1)14.11.1....,

.4 I

1 A 1,-1I

. WI-

kt.k

1 ,-,

1.1'11 FS
`..Ani..1 I- 1.1

[Ili

I 0.

I

1-

I.

1

1111 t .

r.

1.1

i,. A ..

F.1F.

ilAl A
2

I.
4

,

6

1

2

1

",

2

0..

1

0

7

1.

1

2 16,
I

I

1

/-

11, I-.

14 /.
1,1 r.4.

2

164.11

1

1f1

Ik.

1

11

1

17
14
16

A

1

2

1
1

1

7

1

17

1

1

11,

4

1

16,.64.
16 0,1 .

1

it,

k

22
24

2

7

2.4

2'

1

,A1'w.

1111.

1

2P

A

21)

k

Ar

1

1

,,A,

7

'41

7

1 441 I.

2

1

-1,1/.

1

n 14p,A4.

4

A4
.4'

1%,..i

2

1

Vi

41

1

1

1

27

h.f.4

.1., )6.

1

k
4
k

21.

k44h
,t 4 )u.

2
2

1

21

11-, tk.

2

Al

2

2 1646.

1

AA

1

1

47

h

AP
AU

A

n 141,h.
1

9

I

1

1

41

1

62

2

6,1

2

44

h

1

4

117

1,

41+

11,6,1.
16.4)7.

2
1

?

4/1

144hu.
16,416.

1

144

2

0.11,11.

- 44/.

1

1

11110.

n

LI

kr,

1

1

1104)k.

WI 1A

PlIF(*.i
1/4; AMP !IAD.%

1,(117MA1(41(.214)
11.1,11S10A11-

SAwl-:

CWIS"I.AP

PH,
(1,11J

.11,11_11wo

vAPIA1.1kC

rAII-1411,1F". n
; 4 rkw1,41-.

2

Cflp041-A- C111.1,-1, .-.A[101141.,.,
c10,141.-1-

1

CA0...11411-1/4.

4 4

'1

VAri-c
ST(11,

FILMED FROM BEST AVAILABLE COPY

1/70

c9


FREQUENCY DISTRIBUTION TABLE
WITH CHI-SQUARE
SAMPLE PROBLEM FOR STPAC PROGRAM FAKCS

EACH CELL OF THE TABLE CONTAINS AN ACTUAL FREQUENCY ON THE FIRST LINE

<table>
<thead>
<tr>
<th>CATEGORY OF VARIABLE</th>
<th>V</th>
<th>C</th>
<th>1</th>
<th>2</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>C</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>C</td>
<td>2</td>
<td>7</td>
<td>7</td>
<td>16</td>
</tr>
<tr>
<td>2</td>
<td>C</td>
<td>1</td>
<td>5</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>C</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>C</td>
<td>0</td>
<td>5</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>C</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>8</td>
</tr>
</tbody>
</table>

THE VALUE OF CHI-SQUARE FOR 10 DEGREES OF FREEDOM IS EQUAL TO 13.76
THE PROBABILITY OF EXCEEDING THIS VALUE OF CHI-SQUARE BY CHANCE IS 0.184
### Frequency Distribution Table

**Illustrate same data option**

**Preliminary Table**

<table>
<thead>
<tr>
<th>Variable</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>College</td>
<td>1</td>
<td>2</td>
<td>7</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>4</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>3</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>2</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>Column Total</td>
<td>5</td>
<td>28</td>
<td>17</td>
<td>50</td>
</tr>
</tbody>
</table>

The following row categories were ignored - \( C_1 \).

The following column categories were ignored - \( C_2 \).

The following row categories were combined to form category 1 - 1, 2.

The following row categories were combined to form category 3 - 3, 4, 5.
PERCENT-FREQUENCY DISTRIBUTION TABLE
WITH CHI-SQUARE
ILLUSTRATE SAME DATA OPTION

Each cell of the table contains a percentage frequency on the first line and an actual frequency on the second line. Percentages are based on row totals.

<table>
<thead>
<tr>
<th>Categories of Variable</th>
<th>1</th>
<th>2</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 College</td>
<td>52.2</td>
<td>47.8</td>
<td>51.1</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>11</td>
<td>23</td>
</tr>
<tr>
<td>3</td>
<td>72.1</td>
<td>27.9</td>
<td>48.9</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>6</td>
<td>22</td>
</tr>
</tbody>
</table>

The value of chi-square for 1 degree of freedom is equal to 1.241. Yates correction factor was used in obtaining this value of chi-square. The probability of exceeding this value of chi-square by chance is 0.265.
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 5 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead

\[ = 0.55 + \text{network overhead} \]
DESCRIPTIVE TITLE  Statistical Summary
CALLING NAME  (STPAC) STSUM
INSTALLATION NAME  The Pennsylvania State University Computation Center
AUTHOR(S) AND AFFILIATION(S)  Nancy C. Daubert
The Pennsylvania State University Computation Center
LANGUAGE  FORTRAN IV
COMPUTER  IBM System 360/67
PROGRAM AVAILABILITY  Decks and listings presently available
CONTACT  William H. Verity, 104 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

FUNCTIONAL ABSTRACT

STSUM computes, in double-precision arithmetic, the sample size, total, mean, standard deviation, population-variance estimate, standard error, sum of squares, coefficient of variation, third and fourth moments about the origin, third and fourth central moments, alpha (square root of beta 1), alpha 4 (beta 2), momental skewness, and kurtosis. Missing data will be permitted.

Equations

\[
\begin{align*}
V_1 &= \frac{\Sigma X}{N} & \text{moments about origin} \\
V_2 &= \frac{\Sigma X^2}{N} \\
V_3 &= \frac{\Sigma X^3}{N} \\
V_4 &= \frac{\Sigma X^4}{N} \\
\mu_2 &= V_2 - \frac{V_1^2}{N} \\
\mu_3 &= V_3 - 3V_1V_2 + 2V_1^3 \\
\mu_4 &= V_4 - 4V_1V_3 + 6V_1^2V_2 - 3V_1^4 \\
\alpha_3 &= \frac{\sqrt{\beta_1}}{\mu_2} = \left(\frac{\mu_3^2}{\mu_2^3}\right)^{1/2} \\
\alpha_4 &= \beta_2 = \mu_4/\mu_2 \\
\alpha_3/2 &= \text{momental skewness} \\
(\alpha_4 - 3)/2 &= \text{kurtosis} \\
\Sigma X^2 - (\Sigma X)^2/N &= \text{sum of squares}
\end{align*}
\]

continued
REFERENCES

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute, Variables, and Format Cards are essential to STSUM. Begin- and End-Data Cards also are necessary if the data have not already been read for a previous STPAC program. All other STPAC Control Cards are optional.

Missing-Data Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-12</td>
<td>MISSING DATA</td>
</tr>
<tr>
<td>14-21</td>
<td>x, any number, up to eight digits including a decimal point, that will be read as a code for missing data.</td>
</tr>
</tbody>
</table>

Include this card only if some data are missing. If this card is omitted, it will be assumed that all fields contain data (blanks will be read as zero). If this card is included and Cols. 14-21 are blank, zeros and blanks in the Data Cards will be treated as missing data. If Cols. 14-21 are punched, the number x in a data field will be treated as a case of missing data.

Example (starting in Col. 1)

MISSING DATA 99.9

This would provide for a case where Data Cards can contain instances of zero observations and instances of missing data. All data fields previously blank would then be coded 99.9 on the Data Cards.

Description of Input Data

Data must be punched according to the format given on the Format Card. A case and sequence number should be punched on each card (but only if required, as described in the STPAC writeup), as well as the data points in F or E format. The number of a variable is determined by the order in which it is read, which is determined by the format supplied.
Description of Output

Statistics are printed in two tables. All variables are numbered and also are named according to the Variable-Name(s) Card, if it is included.

REFERENCES

SAMPLE INPUT

C

1/70
<table>
<thead>
<tr>
<th>Variable</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.234</td>
<td>0.743</td>
<td>0.567</td>
<td>0.356</td>
</tr>
<tr>
<td>Median</td>
<td>0.203</td>
<td>0.753</td>
<td>0.567</td>
<td>0.356</td>
</tr>
<tr>
<td>Mode</td>
<td>0.234</td>
<td>0.743</td>
<td>0.567</td>
<td>0.356</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.203</td>
<td>0.753</td>
<td>0.567</td>
<td>0.356</td>
</tr>
<tr>
<td>Sum</td>
<td>0.527</td>
<td>1.486</td>
<td>1.134</td>
<td>0.712</td>
</tr>
<tr>
<td>Product</td>
<td>0.052</td>
<td>1.486</td>
<td>1.134</td>
<td>0.712</td>
</tr>
<tr>
<td>Variance</td>
<td>0.020</td>
<td>0.567</td>
<td>0.356</td>
<td>0.234</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>-0.103</td>
<td>-0.100</td>
<td>-0.100</td>
<td>-0.193</td>
</tr>
<tr>
<td>Skewness</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
COST ESTIMATE
For the job listed on the Sample Input, the total running time was 4 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
= $0.44 + network overhead

CONTENTS—(STPAC) STSUM

<table>
<thead>
<tr>
<th>pages</th>
<th>Identification &amp; Abstract</th>
<th>User Instructions</th>
<th>I/O</th>
<th>Cost—Contents</th>
</tr>
</thead>
</table>
T Test on Difference between Means

(STPAC) TTEST

The Pennsylvania State University Computation Center

Nancy C. Daubert
The Pennsylvania State University Computation Center

FORTRAN IV

IBM System 360/67

Decks and listings presently available

William H. Verity, 104 Computer Building,
The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

TTEST variables (samples) may be correlated or noncorrelated. Missing observations are permitted. Data may be submitted in the form of a single deck or multiple decks. Output can consist of values of t, frequencies, probabilities, and levels of significance. TTEST will perform a t test on each of all possible pairs of variables, each t being computed independently of all others.

For a t test on the difference between means of correlated samples, the formula used is

\[ t = \frac{\sum D/N}{\left\{\sum D^2 - \frac{(\sum D)^2}{N}\right\}/[N(N-1)]}^{1/2} \]

where \( N \) is the number of pairs and \( D \) is the difference between the scores for each pair. The number of degrees of freedom is expressed by \( N - 1 \).

For a t test on the difference between means of noncorrelated samples, with separate variances, the formula used is

\[ t = (\bar{X}_1 - \bar{X}_2)/(S_{X_1}^2 + S_{X_2}^2)^{1/2}, \]

continued
where the $X$'s are the means and the $S_{X_1}^2$ and $S_{X_2}^2$ are the squares of the standard errors of the means.

If the frequencies of the two samples are equal, the number of degrees of freedom are equal to the frequency minus 1. If the frequencies are unequal, the program uses the lowest frequency minus 1 as the number of degrees of freedom.

A t test on the difference between means of noncorrelated samples, with pooled variances, involves the formula

$$t = (\bar{X}_1 - \bar{X}_2)/\left[S^2\left(\frac{1}{K_1} + \frac{1}{K_2}\right)\right]^{1/2},$$

where $S^2$ is the variance—i.e., the sum of squares has been calculated by summing the squared deviations for each individual case from the mean of the group in which that case is found—and the $K$'s are the individual group frequencies. The number of degrees of freedom is equal to $K_1 + K_2 - 2$.

REFERENCES

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute, Variables, and Format Cards are essential to TTEST. Begin- and End-Data Cards also are necessary if the data have already been read for a previous STPAC program. All other STPAC Control Cards are optional. There are in addition some required and optional Control Cards specific to TTEST; these are described below.

Correlated or Noncorrelated Card (one required)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–10</td>
<td>CORRELATED</td>
</tr>
<tr>
<td></td>
<td>or</td>
</tr>
<tr>
<td>1–13</td>
<td>NONCORRELATED</td>
</tr>
<tr>
<td></td>
<td>or</td>
</tr>
<tr>
<td>1–20</td>
<td>NONCORRELATED POOLED</td>
</tr>
</tbody>
</table>

This card informs the program as to whether the variables are correlated or uncorrelated (noncorrelated), and, if the latter, whether pooled variance or unpooled variance is to be used.

Punch Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–5</td>
<td>PUNCH</td>
</tr>
</tbody>
</table>

This card should be included only if the values of t are to be punched on cards in addition to being printed.

Decks Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–7</td>
<td>DECKS M</td>
</tr>
<tr>
<td></td>
<td>or</td>
</tr>
<tr>
<td>1–5</td>
<td>DECKS</td>
</tr>
</tbody>
</table>

This card is one of these.
This card should be included if the data should be submitted in multiple decks rather than in a single deck. DECKS M is used if all decks have equal numbers of variables punched on them and M represents that number of variables. DECKS is used when decks have different numbers of variables punched on them; in this case, the number of variables per card is included in the data. See Description of Input Data, below.

Probabilities Card (optional)

Columns  Contents
1-13  PROBABILITIES

This card causes a table of probabilities for each t to be printed.

Significance-Levels Card (optional)

Columns  Contents
1-19  SIGNIFICANCE LEVELS

Include this card to print out a table containing entries that describe significance at the 0.05, 0.025, 0.01, and 0.001 levels.

Missing-Data Card (optional)

Columns  Contents
1-12  MISSING DATA
14-21  Any number up to eight digits, including a decimal point, that will be read as a code for missing data.

Include this card only if some data are missing. If this card is omitted, it will be assumed that all fields contain data (blanks will be read as zero). If this card is included and Cols. 14-21 are blank, zeros and blanks in the Data Cards will be treated as missing data. If Cols. 14-21 are punched, the number x in a data field will be treated as a case of missing data.

Example (starting in Col. 1)
MISSING DATA 99.9

This would provide for a case where Data Cards can contain instances of zero observations and instances of missing data. All data fields previously blank would then be coded 99.9 on the Data Cards.

Description of Input Data

Data must be punched according to the format given on the Format continued
Card. In standard form, each card should contain case and sequence numbers in I format (but only if required for multiple cards per case, as described in the STPAC writeup), followed by data points for each variable in F or E format.

If the DECKS M option (1st form) is used, each card should contain a deck-identification number in I format and a data point in F, E, or D format for each variable in that deck.

If the DECKS option (2nd form) is used, each card should contain not only a deck-identification number but also the number of variables on that card (i.e., that deck), both in I format, in addition to a data point in F, E, or D format for each variable in that deck. All cards for one deck—containing the same deck-identification number and the same number of variables on every card—must be together. Correlated variables (samples) will not be handled with the DECKS option.

The number of a variable is determined by the order in which it is read, which is determined by the format supplied (and the order of the decks if multiple decks are used). In the case of decks, where the first deck contains three variables and the second contains four, the variables will be renumbered from 1 through 7.

Description of Output

Values of t will be printed and also punched at the user's option. In the case of correlated samples, a table of frequencies also will be printed. Tables of probabilities of significance levels will be printed at the user's option.

All tables are printed as triangular matrices. Every variable is numbered and named according to the Variable-Name(s) Card if it is included.

REFERENCES

SAMPLE INPUT

1 1 1 1
2 1 1 2
3 1 1 3
4 1 1 4
5 1 1 5
6 1 1 6
7 1 1 7
8 1 1 8
9 1 1 9
10 1 1 10
11 1 1 11
12 1 1 12
13 1 1 13
14 1 1 14
15 1 1 15
16 1 1 16
17 1 1 17
18 1 1 18
19 1 1 19
20 1 1 20
21 1 1 21
22 1 1 22
23 1 1 23
24 1 1 24
25 1 1 25
26 1 1 26
27 1 1 27
28 1 1 28
29 1 1 29
30 1 1 30
31 1 1 31
32 1 1 32
33 1 1 33
34 1 1 34
35 1 1 35
36 1 1 36
37 1 1 37
38 1 1 38
39 1 1 39
40 1 1 40
41 1 1 41
42 1 1 42
43 1 1 43
44 1 1 44
45 1 1 45
46 1 1 46
47 1 1 47
48 1 1 48
49 1 1 49
50 1 1 50
51 1 1 51
52 1 1 52
53 1 1 53
54 1 1 54
55 1 1 55
56 1 1 56
57 1 1 57
58 1 1 58
59 1 1 59
60 1 1 60
61 1 1 61
62 1 1 62
63 1 1 63
64 1 1 64
65 1 1 65
66 1 1 66
67 1 1 67
68 1 1 68
69 1 1 69
70 1 1 70

FILMED FROM BEST AVAILABLE COPY
SAMPLE OUTPUT

TEST OF SECOND FISHER DECK

VARIABLE   MEAN  STANDARD FISHER SQUARED

1          14  2.187440 03  0.173640 03

2          14  0.239403 03  0.210520 03

TEST OF SECOND FISHER DECK

T TEST

UNCORRELATED SAMPLES

1

NO. OBS.  14

2          14  7.72
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 4 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
               = $0.44 + network overhead

CONTENTS—(STPAC) TTEST

<table>
<thead>
<tr>
<th>pages</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>Identification &amp; Abstract</td>
</tr>
<tr>
<td>3-5</td>
<td>User Instructions</td>
</tr>
<tr>
<td>7-8</td>
<td>I/O</td>
</tr>
<tr>
<td>9</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>
PPMCR computes not only a Pearson product-moment correlation coefficient for each possible pair of input variables, but also the mean and standard deviation of each input variable.
USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

PPMCR can have an unlimited number of observations per variable. The output can be punched and then submitted at a later date as input to another program, such as regression analysis, or it can be passed directly to another program in STPAC. All computations use double-precision arithmetic.

Control Cards and Their Functions

PPMCR has one optional Control Card in addition to those applicable to all STPAC programs. There is no standard format for this program, so a Format Card must be supplied. Also, a Variables Card is required.

Punch Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>PUNCH</td>
</tr>
</tbody>
</table>

This card will cause PPMCR to punch a deck that can be used later as input to another STPAC program [for example, DNREG, EIN No. 000 0051(h)]. Since intermediate results can be passed directly to another STPAC program, the use of punched-card output is recommended only where it is desirable to break up the run.

Description of Input Data

Data Cards must be supplied to PPMCR, along with the proper format. If, as discussed in the STPAC writeup, case and sequence numbers are used (only when all the variables do not fit on one card), the format must allow for them. Data are read by using an F- or E-format code (both for single-precision floating-point numbers). Case and sequence numbers, if used, are read with an I-format code.

Description of Output

The first page of PPMCR output is a table of means and standard deviations for each of the input variables. Then, starting on a new page, the correlation coefficients are printed in the form of a triangular matrix, with the rows and columns identified by variable...
number and name if a Variable-Name(s) Card was used.

If punched output was requested, the punched deck contains

(1) Title Card
(2) A set of Variable-Name(s) Cards, if the variables were named
(3) A Begin-Data Card
(4) Cards containing the means, standard deviations, and correlation coefficients and a card with the number of variables and number of observations
(5) An End-Data Card
SAMPLE INPUT

EXECUTE PDCR
TITLE SAMPLE FOR EXAMPLE ON PUNCH 414
VARIABLES 3
VARIABLE NAMES 1=XX2.X3X4Y
FORMAT (F3.1,F2.0,F4.2)
BEGIN DATA
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
004244
END DATA
EXECUTE SIGPP
INPUT FROM PUNCH
LEVEL .1 T1N
EXECUTE LDRREG
INPUT FROM PUNCH
PERFORM Y
PUNCH 123444
EXECUTE LDRREG
INPUT FROM PUNCH
PERFORM Y
EXECUTE CANON
INPUT FROM PUNCH
LEFT V
RIGHT 123444
EXECUTE FANAL
INPUT FROM PUNCH
EXIT V
EXECUTE VARMX
INPUT FROM FANAL
STOP
SAMPLE OUTPUT

MEANS AND STANDARD DEVIATIONS

<table>
<thead>
<tr>
<th></th>
<th>X1</th>
<th>11.94</th>
<th>STO. DEV</th>
<th>1.15</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>X2</td>
<td>42.11</td>
<td>13.47</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Y</td>
<td>81.28</td>
<td>27.06</td>
<td></td>
</tr>
</tbody>
</table>

CORRELATION COEFFICIENTS

<table>
<thead>
<tr>
<th></th>
<th>X1</th>
<th>X2</th>
<th>X2</th>
<th>Y</th>
<th>0.4616</th>
<th>0.3545</th>
</tr>
</thead>
</table>

TOTAL NUMBER OF DATA CARDS PROCESSED: 18
NUMBER OF VARIABLES EXAMINED: 3
NUMBER OF OBSERVATIONS: 19
COST ESTIMATE

For the job listed on the Sample Input, the total running time on the central processor unit was 6 seconds, at $0.11 per second.

Charge to user = computer time + network overhead
               = $0.66 + network overhead

CONTENTS—(STPAC) PPMCR

pages
  1  Identification & Abstract
  3– 4  User Instructions
  5– 6  I/O
  7  Cost—Contents
**DESCRIPTIVE TITLE**
Significance of Pearson Product-Moment Correlation Coefficient

**CALLING NAME**
(STPAC) SIGPP

**INSTALLATION NAME**
The Pennsylvania State University Computation Center

**AUTHOR(S) AND AFFILIATION(S)**
- W.M. Stiteler, III
  Mathematics Department, The Pennsylvania State University
- F. Yates Borden
  Forestry Department, The Pennsylvania State University
- H.D. Knoble, The Pennsylvania State University Computation Center

**LANGUAGE**
FORTRAN IV

**COMPUTER**
IBM System 360/67

**PROGRAM AVAILABILITY**
Decks and listings presently available

**CONTACT**
William H. Verity, 104 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

**FUNCTIONAL ABSTRACT**
SIGPP is to be used optionally in conjunction with the correlation programs to determine the significance of correlation coefficients at any desired probability levels. That is, the null hypothesis $r = 0$ (alternate: $r \neq 0$) is tested for each correlation coefficient.

**REFERENCES**
USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

In addition to Control Cards specified for STPAC, the following Control Card may be used for this program.

Level Card(s) (optional)

Columns | Contents
---|---
1–80 | LEVEL  RRRRRRRR  CCCC

There may be from zero to five of these cards for a problem. LEVEL is a key word. RRRRRRRR is a two- to eight-character floating-point field, including a decimal point (e.g., .001 or .05), indicating the probability of making a type I error (rejecting the null hypothesis when it is true). 0 < RRRRRRRR < 1. CCCC is a one- to four-character symbol to be associated with RRRRRRRR (e.g., * or .05).

Description of Input Data

Punched-card output from a correlation program may be used as data. That is,

BEGIN DATA
...
Output from a correlation program
...
END DATA

By using the Input-From Card as described in the STPAC writeup, SIGPP may be executed during the same run as a correlation program.

To repeat a problem using the same data but different Level-Card options, the Same-Data Card may be used according to the STPAC rules.

NOTES:

1. Of the above three options (Punch Card, Input-From Card, Same-Data Card), one must be chosen for a problem.
2. Although means and standard-deviations cards are not necessary for SIGPP, these may be left intact on punched-card input as they will automatically be ignored by this program.

3. Level Card(s), if used, may be placed anywhere after the Execute Card and before the Begin-Data Card if the latter card is present.

The approximations in the program begin to lose accuracy as the number of observations becomes less than 10. According to Ref. 1, the approximations in SIGPP are valid also for Spearman rank coefficients for cases where there are more than eight ranks.

Description of Output

A probability of making a type I error is computed for each correlation coefficient, using a two-tailed test based on a transformation and approximation of the Z (standard normal) probability density function.

If Level Card(s) have been used, the null hypothesis $H_0: r = 0$ is tested versus the alternate $H_0: r \neq 0$ at the specified level(s) of significance and a triangular matrix of results is printed, using corresponding symbols. The sign of the correlation coefficient also is printed with each respective symbol. Thus, each position of the triangular matrix of correlations will have a sign and a symbol that designate the sign and significance of the correlation coefficient of that corresponding position. All probabilities greater than the greatest probability on all of the Level Card(s) appear in the output as #NS#, meaning that the null hypothesis is to be accepted at the level of significance greater than or equal to the largest probability specified.

If no Level Card(s) have been used for a problem, each position of the triangular matrix will have a sign and the actual probability (of making a type I error or rejecting the null hypothesis when it is true) associated with the correlation coefficient of that corresponding position.

REFERENCES

**SAMPLE OUTPUT**

<table>
<thead>
<tr>
<th>X1</th>
<th>X2</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>TEN</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>#NS#</td>
</tr>
</tbody>
</table>

**NOTE:**

- The sign accompanying a probability symbol is the sign of the respective correlation coefficient.
- Probabilities less than or equal to 0.10000 are designated by #NS# indicating acceptance of the null hypothesis at this level.
- Probabilities greater than 0.10000 are designated by TEN indicating rejection of the null hypothesis at this level.
- Number of observations = 10
- Degrees of freedom = 16

---

**THE NULL HYPOTHESIS: R .EQ. 0 VS. THE ALTERNATE: R .NE. 0**

- Each correlation coefficient, R, is tested for significance.

**NOTE:**

- The null hypothesis is tested against the alternative hypothesis at each level.
COST ESTIMATE

For the job listed on the Sample Input, the total running time on the central processor unit was 6 seconds, at $0.11 per second.

Charge to user = computer time + network overhead
= $0.66 + network overhead

CONTENTS—(STPAC) SIGPP

<table>
<thead>
<tr>
<th>pages</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Identification &amp; Abstract</td>
</tr>
<tr>
<td>3-4</td>
<td>User Instructions</td>
</tr>
<tr>
<td>5-6</td>
<td>I/O</td>
</tr>
<tr>
<td>7</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>
USER INSTRUCTIONS
This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

PARCOR can have an unlimited number of observations per variable. Up to 27 variables may be specified on the Include and Delete Cards for the calculation of partial correlations and multiple correlations. All computations are done in double-precision arithmetic.

Control Cards and Their Functions
There is no standard format for this program, so a Format Card must be supplied. A Variable Card is also required, and a Variable-Names Card may be included.

PARCOR has four types of optional Control Cards in addition to those applicable to all programs in STPAC.

Print Card (optional)

Columns Contents
1 5 PRINT

This card instructs the program to print the variance-covariance matrix for the total number of variables.

All Card (optional)

Columns Contents
1 3 ALL

This card specifies that partial correlations and multiple correlations of all the variables are to be computed, using the entire variance-covariance matrix.

Include Cards (optional)

Columns Contents
1 7 INCLUDE
9-80 i j k ...

This card instructs the program to compute partial correlation and multiple correlations for variables i, j, k, ..., only. The variable list specified for a single calculation must fit on
one card. The variables may be specified by their number or, if a Variable-Names Card was supplied, by their names. They must be separated by at least one blank, and hence at most 27 variables could be specified for a single calculation.

Delete Card(s) (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-6</td>
<td>DELETE</td>
</tr>
<tr>
<td>8-80</td>
<td>i j k ...</td>
</tr>
</tbody>
</table>

This card instructs the program to delete variables i, j, k, ..., from the calculation of partial correlations and multiple correlations. All variables specified for a single calculation must fit on one card. They may be specified by number or by name (if a Variable-Names Card was supplied), and they must be separated by one or more blanks. Up to 27 variables may be specified on each Delete Card.

Example: DELETE AGE SEX WEIGHT HEIGHT

This card specifies that all the variables except age, sex, weight, and height should be included in the calculation of partial and multiple correlations. In this example, the variables are specified by name, and it is assumed that a Variable-Names Card was supplied.

NOTE: PARCOR Control Cards may be inserted anywhere among the STPAC Control Cards and they may appear in any order. A given set of Control Cards may consist of any number of Include or Delete Cards and, at most, one Print Card and one All Card for any given set of data.

Description of Input Data

Data Cards must be supplied to PARCOR along with the appropriate Format Card. If, as described in the STPAC writeup, the case and sequence numbers are used (i.e., if all the variables do not fit on one card), the format must allow for them. Data are read using F- or E-format code, and the case and sequence numbers, if used, with an I-format code.

Description of Output

If a Print Control Card was supplied, the variance-covariance matrix for all the variables is printed in the form of a triangular matrix, together with the means of all the variables. The rows and columns are identified by variable numbers and, if a Variable-Names Card was supplied, by their names. continued
For each Include, Delete, and All Card, a triangular matrix of partial correlations of the specified variables is printed. The rows and columns are identified by the variable numbers and, if a Variable-Names Card was included, by their names. The heading gives the total number of variables as well as the number of variables included in the calculation of partial correlations and multiple correlations. The multiple correlations of these variables are printed below the matrix.
### VARIANCE-COVARIANCE MATRIX

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>VAR1</td>
<td>7.21E-01</td>
<td>8.16E-01</td>
<td>6.22E-01</td>
<td>7.35E-01</td>
<td>7.38E-01</td>
<td>6.39E-01</td>
<td>5.17E-01</td>
<td>7.20E-01</td>
<td>6.39E-01</td>
<td>6.72E-01</td>
<td>5.17E-01</td>
<td>4.71E-01</td>
</tr>
<tr>
<td>VAR3</td>
<td>6.22E-01</td>
<td>7.22E-01</td>
<td>5.87E-01</td>
<td>7.00E-01</td>
<td>7.03E-01</td>
<td>6.04E-01</td>
<td>4.82E-01</td>
<td>6.93E-01</td>
<td>6.04E-01</td>
<td>6.47E-01</td>
<td>4.82E-01</td>
<td>4.45E-01</td>
</tr>
<tr>
<td>VAR4</td>
<td>7.35E-01</td>
<td>8.35E-01</td>
<td>7.00E-01</td>
<td>8.18E-01</td>
<td>8.21E-01</td>
<td>7.22E-01</td>
<td>6.00E-01</td>
<td>8.13E-01</td>
<td>7.22E-01</td>
<td>7.64E-01</td>
<td>6.00E-01</td>
<td>5.61E-01</td>
</tr>
<tr>
<td>VAR5</td>
<td>7.38E-01</td>
<td>8.38E-01</td>
<td>7.03E-01</td>
<td>8.21E-01</td>
<td>8.24E-01</td>
<td>7.25E-01</td>
<td>6.03E-01</td>
<td>8.14E-01</td>
<td>7.25E-01</td>
<td>7.68E-01</td>
<td>6.03E-01</td>
<td>5.62E-01</td>
</tr>
<tr>
<td>VAR7</td>
<td>5.17E-01</td>
<td>6.17E-01</td>
<td>4.82E-01</td>
<td>6.00E-01</td>
<td>6.03E-01</td>
<td>5.04E-01</td>
<td>3.82E-01</td>
<td>5.73E-01</td>
<td>5.04E-01</td>
<td>5.47E-01</td>
<td>3.82E-01</td>
<td>3.42E-01</td>
</tr>
<tr>
<td>VAR9</td>
<td>5.17E-01</td>
<td>6.17E-01</td>
<td>4.82E-01</td>
<td>6.00E-01</td>
<td>6.03E-01</td>
<td>5.04E-01</td>
<td>3.82E-01</td>
<td>5.73E-01</td>
<td>5.04E-01</td>
<td>5.47E-01</td>
<td>3.82E-01</td>
<td>3.42E-01</td>
</tr>
<tr>
<td>VAR12</td>
<td>6.22E-01</td>
<td>7.22E-01</td>
<td>5.87E-01</td>
<td>7.00E-01</td>
<td>7.03E-01</td>
<td>6.04E-01</td>
<td>4.82E-01</td>
<td>6.93E-01</td>
<td>6.04E-01</td>
<td>6.47E-01</td>
<td>4.82E-01</td>
<td>4.45E-01</td>
</tr>
<tr>
<td>VAR13</td>
<td>7.35E-01</td>
<td>8.35E-01</td>
<td>7.00E-01</td>
<td>8.18E-01</td>
<td>8.21E-01</td>
<td>7.22E-01</td>
<td>6.00E-01</td>
<td>8.14E-01</td>
<td>7.22E-01</td>
<td>7.64E-01</td>
<td>6.00E-01</td>
<td>5.61E-01</td>
</tr>
<tr>
<td>VAR14</td>
<td>7.38E-01</td>
<td>8.38E-01</td>
<td>7.03E-01</td>
<td>8.21E-01</td>
<td>8.24E-01</td>
<td>7.25E-01</td>
<td>6.03E-01</td>
<td>8.14E-01</td>
<td>7.25E-01</td>
<td>7.68E-01</td>
<td>6.03E-01</td>
<td>5.62E-01</td>
</tr>
<tr>
<td>VAR16</td>
<td>5.17E-01</td>
<td>6.17E-01</td>
<td>4.82E-01</td>
<td>6.00E-01</td>
<td>6.03E-01</td>
<td>5.04E-01</td>
<td>3.82E-01</td>
<td>5.73E-01</td>
<td>5.04E-01</td>
<td>5.47E-01</td>
<td>3.82E-01</td>
<td>3.42E-01</td>
</tr>
<tr>
<td>VAR18</td>
<td>5.17E-01</td>
<td>6.17E-01</td>
<td>4.82E-01</td>
<td>6.00E-01</td>
<td>6.03E-01</td>
<td>5.04E-01</td>
<td>3.82E-01</td>
<td>5.73E-01</td>
<td>5.04E-01</td>
<td>5.47E-01</td>
<td>3.82E-01</td>
<td>3.42E-01</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEANS</td>
<td>2.23E-01</td>
<td>4.44E-01</td>
<td>4.06E-01</td>
<td>2.39E-01</td>
<td>2.90E-01</td>
<td>8.14E-01</td>
<td>3.57E-01</td>
<td>4.70E-01</td>
<td>3.57E-01</td>
<td>3.86E-01</td>
<td>3.57E-01</td>
<td>2.86E-01</td>
</tr>
</tbody>
</table>
### Partial Correlation Coefficients Matrix

**Total Number of Variables**: 30  
**Number of Variables in Correlation**: 12

<table>
<thead>
<tr>
<th></th>
<th>VAR19</th>
<th>VAR20</th>
<th>VAR21</th>
<th>VAR22</th>
<th>VAR23</th>
<th>VAR24</th>
<th>VAR25</th>
<th>VAR26</th>
<th>VAR27</th>
<th>VAR28</th>
<th>VAR29</th>
<th>VAR30</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1.1128</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>-0.0054</td>
<td>-0.0033</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>0.2069</td>
<td>-0.2327</td>
<td>-0.1581</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>0.1556</td>
<td>-0.2559</td>
<td>-0.0343</td>
<td>0.0033</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>-0.0675</td>
<td>-0.1240</td>
<td>0.2081</td>
<td>-0.0760</td>
<td>0.0075</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>0.2013</td>
<td>0.1904</td>
<td>-0.1581</td>
<td>-0.2052</td>
<td>0.0725</td>
<td>-0.0014</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>0.1642</td>
<td>0.1904</td>
<td>0.3793</td>
<td>0.2712</td>
<td>0.0212</td>
<td>-0.0075</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>0.1437</td>
<td>-0.0112</td>
<td>0.1581</td>
<td>0.1645</td>
<td>0.1702</td>
<td>-0.0001</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>0.0717</td>
<td>0.3640</td>
<td>0.3163</td>
<td>0.4593</td>
<td>0.0063</td>
<td>0.1972</td>
<td>-0.0392</td>
<td>0.0090</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>-0.0226</td>
<td>0.2430</td>
<td>0.2444</td>
<td>0.1581</td>
<td>0.1695</td>
<td>0.0849</td>
<td>-0.0813</td>
<td>0.0788</td>
<td>0.0908</td>
<td>1.0000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>-0.0135</td>
<td>0.2667</td>
<td>0.1517</td>
<td>0.3432</td>
<td>0.0179</td>
<td>-0.0589</td>
<td>0.2429</td>
<td>-0.0053</td>
<td>0.2324</td>
<td>1.0000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Multiple Correlation Coefficients

<table>
<thead>
<tr>
<th></th>
<th>VAR19</th>
<th>VAR20</th>
<th>VAR21</th>
<th>VAR22</th>
<th>VAR23</th>
<th>VAR24</th>
<th>VAR25</th>
<th>VAR26</th>
<th>VAR27</th>
<th>VAR28</th>
<th>VAR29</th>
<th>VAR30</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>0.5104</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>0.4864</td>
<td>21.4527</td>
<td>0.6517</td>
<td>0.6413</td>
<td>0.4048</td>
<td>0.7366</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>0.5443</td>
<td>0.8045</td>
<td>0.4939</td>
<td>0.5402</td>
<td>0.7576</td>
<td>0.7619</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### PARTIAL CORRELATION COEFFICIENTS MATRIX

**REAL NUMER OF VARIABLES = 30**  
**NUMBER OF VARIABLES IN CORRELATION = 17**

<table>
<thead>
<tr>
<th></th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>VAR4</td>
<td>VAR15</td>
<td>VAR16</td>
<td>VAR17</td>
<td>VAR18</td>
<td>VAR19</td>
<td>VAR20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>C.065C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>C.052C</td>
<td>C.030C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>C.0120</td>
<td>C.0435</td>
<td>C.0955</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>0.7175</td>
<td></td>
<td>C.0372</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>C.2717</td>
<td></td>
<td>C.0834</td>
<td>C.0381</td>
<td>C.0572</td>
<td>C.0369</td>
<td>C.0370</td>
<td>C.0370</td>
<td>C.0370</td>
<td>C.0370</td>
<td>C.0370</td>
<td>C.0370</td>
<td>C.0370</td>
<td>C.0370</td>
<td>C.0370</td>
<td>C.0370</td>
<td>C.0370</td>
</tr>
<tr>
<td>21</td>
<td>0.0214</td>
<td>C.0265</td>
<td>C.1695</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
<td>C.0376</td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
<td>C.0372</td>
</tr>
<tr>
<td>24</td>
<td>-0.0712</td>
<td>C.0215</td>
<td>C.2123</td>
<td>C.0547</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
<td>C.0373</td>
</tr>
<tr>
<td>28</td>
<td>0.0312</td>
<td>C.0368</td>
<td>C.0155</td>
<td>C.0647</td>
<td>C.0405</td>
<td>C.0513</td>
<td>C.0134</td>
<td>C.0283</td>
<td>C.0293</td>
<td>C.0401</td>
<td>C.0283</td>
<td>C.0283</td>
<td>C.0283</td>
<td>C.0283</td>
<td>C.0283</td>
<td>C.0283</td>
<td>C.0283</td>
</tr>
<tr>
<td>29</td>
<td>0.2209</td>
<td>C.0143</td>
<td>C.0581</td>
<td>C.0375</td>
<td>C.0112</td>
<td>C.0133</td>
<td>C.0042</td>
<td>C.0156</td>
<td>C.0156</td>
<td>C.0156</td>
<td>C.0156</td>
<td>C.0156</td>
<td>C.0156</td>
<td>C.0156</td>
<td>C.0156</td>
<td>C.0156</td>
<td>C.0156</td>
</tr>
</tbody>
</table>

### MULTIPLE CORRELATION COEFFICIENTS

<table>
<thead>
<tr>
<th></th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>C.065C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>C.0405</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>C.0405</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>C.052C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>0.6804</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>C.0120</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### INPUT ERROR

The variable - VAR29 specified on the INCLUDE or DELETE card has not been defined on the VARIABLE-NAMES card.

Program proceeded to the next problem.

Please review the variable specification on the VARIABLE-NAMES card and ensure all variables are correctly defined.

**ERROR**: INCORRECT CONTROL CARDS

The keyword DELETE was not recognized.

Program proceeded to the next card.
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 7 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead

= $0.77 + network overhead

CONTENTS—(STPAC) PARCOR

<table>
<thead>
<tr>
<th>pages</th>
<th>Identification &amp; Abstract</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td></td>
</tr>
<tr>
<td>3-5</td>
<td>User Instructions</td>
</tr>
<tr>
<td>7-10</td>
<td>I/O</td>
</tr>
<tr>
<td>11</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>

4/71  11
FUNCTIONAL ABSTRACT

CANON solves successively for the most significant canonical correlation coefficients. At the same time, two sets of weights associated with each pair of canonical variates are computed.

All matrix computations are done in double-precision floating-point arithmetic. The set of criteria and predictor variables is divided into two groups: 1 (lefthand) and r (righthand) variables, the smaller set automatically becoming the righthand variables. The program then uses a variation of the jacobian method to extract r eigenvalues and r eigenvectors (righthand weights). These are then used through further calculations to compute r canonical correlation coefficients and r sets of lefthand weights. For each canonical correlation coefficient, chi square is computed with the aid of the eigenvalues and the parameters N (the number of observations), r, and 1.

Original variables are brought into the canonical-correlation problem by means of Right- and Left-Variables Cards. Computationally, it is irrelevant whether the variables on the left or on the right are considered as criteria or predictor variables. However, to save computer time, the set with the smallest number of variables (always put into the righthand category by the pro-
gram) should be designated as right. Any or all of the variables brought into the problem may be named by using Name Cards according to STPAC rules.

The maximum output records for a problem may be estimated as follows. Let R be the number of righthand variables, L be the number of lefthand variables, and R < L. Then, the number of records equals R(16 + L) + 10.

Given two sets of variables—not necessarily random variables—[X_i (where i = 1, 2, ..., p, the number of predictor variables, and Y_i (where i = 1, 2, ..., q, the number of criterion variables] the problem that CANON solves is essentially that of maximizing the correlation between certain members of the two sets, simultaneously reducing the others to zero. Canonical correlation, then, may be defined in one sense as a process by which the relationship between two sets of variables is reduced to its simplest form. In canonical correlation, both multiple criteria and multiple predictors can be involved. (Note that, when q = 1 and p > 1, the problem is equivalent to that of multiple regression.)

For a hypothetical example, let p = 3 and q = 2. Here, it is possible to extract two canonical correlation coefficients—say R_1 and R_2. There is a total of p + q = 5 variables—say X_1, X_2, X_3 (predictors) and X_4, X_5 (criteria). There will be two pairs of canonical variates—say U_1V_1 and U_2V_2. Suppose for R_1 that the weights for the predictor variables are P_{11}, P_{12}, P_{13} and for the criteria variables that the weights are C_{11}, C_{12}. Then, the pair of new variates, the canonical variates, which are multiples of a linear function of the original variables, are

\[ U_1 = K_{11}(P_{11}X_1 + P_{12}X_2 + P_{13}X_3), \]
\[ V_1 = K_{12}(C_{11}X_4 + C_{12}X_5), \]

where the K's are arbitrary constants of proportionality and U_1 and V_1 are correlated as described by R_1. A similar situation exists for U_2 and V_2, which are correlated as described by R_2. Furthermore, in general all of the U's and V's have zero mean and unit variance; any U is independent of any other U and likewise for the V's; the correlation between any U and V is zero, except for r correlations R_1, R_2, ..., R_r, which are the correlations between U_1 and V_1, U_2 and V_2, ..., U_r and V_r.

As far as interpretation is concerned, the same difficulty arises as in factor analysis: that of knowing whether or not the linear functions correspond to anything real or whether they are merely mathematical figments.

continued
REFERENCES


USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

In addition to Control Cards specified in the STPAC writeup, the following Control Cards may be used for this program. At least one of each is necessary unless specified as optional. The key word and parameters for each card may be punched anywhere on that card, separated by one or more blanks, with the stipulation that the order of each field be adhered to as follows.

Right-Variables Card(s)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-80</td>
<td>RIGHT III III III ... III</td>
</tr>
</tbody>
</table>

RIGHT is a key word. III is either (1) a field of one- to three-digit variable numbers of those variables to be used as right variables or (2) a field of one- to eight-character variable names of those variables to be used as right variables. In the first case, III must be a number between one and the number of variables that are brought into a problem. In the second case, III must be a variable name as defined on the STPAC Name Card(s). In both cases, there may be as many of these fields per Right-Variables Card(s) as will fit on the card or as are necessary to define the problem. However, a maximum of 50 right variables is allowed. Numbered and named fields may be mixed on one card and between cards.

Examples:

RIGHT 7 11 2 1 4
RIGHT OAT .WHEAT CORN
RIGHT DBH VOLUME PINEHT HRDWDHT 14 17

Left-Variables Card(s)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-80</td>
<td>LEFT III III III ... III</td>
</tr>
</tbody>
</table>

LEFT is a key word. III is defined the same as above, except that its application is to left variables.

continued
Examples:

LEFT 6 3 5 8 9 10 12 21
LEFT RELATIVE ABSOLUTE 'NO. SPOR'
LEFT 13 PROBEL 9 STRESS STRAIN 10

Tolerance Card (optional)

Columns Contents

1–80 TOL RRRRRRR

This card reassigns or overrides the default tolerance to be used when pivoting through the correlation matrix during inversions. The user is cautioned when changing the tolerance, which is set to 0.01 as a default. The tolerance may not be changed to any value outside the range $10^{-1}$–$10^{-7}$ and will be reset to $10^{-2}$ if an attempt is made to do so.

TOL is a key word. RRRRRRRR is a two- to eight-character floating-point field, including a decimal point, designating the tolerance to be used for the current problem.

Description of Input Data

Punched-card output from a Pearson product-moment correlation program may be used as data. That is,

BEGIN DATA

Output from a Pearson product-moment correlation program

END DATA

By use of the Input-From Card as described in the STPAC writeup, CANON may be executed during the same run as a correlation program.

To repeat a problem using the same data but different Right-Variables and/or Left-Variables options, the Same-Data Card may be used according to the STPAC rules.

NOTES:

1. Of the three above options (Punch Card, Input-From Card, Same-Data Card), one must be chosen for a problem.

2. Although means and standard-deviations cards are not necessary...
for CANON, these may be left intact on punched-card input, as they will be ignored by this program.

3. The cards described above may be placed in any order after the Execute Card and before the Begin-Data Card, if one is used.

Description of Output

The normalized and unnormalized vectors of weights associated with the set of criteria variables and the corresponding vectors associated with the set of predictor variables are printed out with each canonical correlation coefficient. The value of chi square, Wilk's lambda, and the probability of obtaining a greater value of chi square associated with the hypothesis that the predictor variate is unrelated to the criteria variate also is printed with each coefficient.

Other pertinent information such as title, variable names, number of observations, degrees of freedom for chi square, and eigenvalues used in calculation of the weights also will be printed.
### Canonical Correlation

#### Eigenvalue

| Eigenvalue 1 | 0.4823127B |

| Sum of Eigenvalues | 0.4823127B |

#### No. Variables in Correlation Array

- No. of Observations: 10
- Chi-Square: 10.7 with 2 Degrees of Freedom
- Wilks Lambda: 0.518
- Squared Canonical Correlation Coefficient: 0.437

#### Normalized Variables

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Variable Number</th>
<th>Normalized Left-Hand Weights</th>
<th>Normalized Right-Hand Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>1</td>
<td>0.999</td>
<td>0.673</td>
</tr>
<tr>
<td>X2</td>
<td>2</td>
<td>0.065</td>
<td>0.046</td>
</tr>
</tbody>
</table>

#### Unnormalized Variables

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Variable Number</th>
<th>Unnormalized Left-Hand Weights</th>
<th>Unnormalized Right-Hand Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>3</td>
<td>1.000</td>
<td>1.000</td>
</tr>
</tbody>
</table>

#### Probabilty of a Greater Value of Chi-Square

- 0.005
COST ESTIMATE

For the job listed on the Sample Input, the total running time on the central processor unit was 6 seconds, at $0.11 per second.

Charge to user = computer time + network overhead
   = $0.66 + network overhead

CONTENTS—(STPAC) CANON

<table>
<thead>
<tr>
<th>pages</th>
<th>Identification &amp; Abstract</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-3</td>
<td></td>
</tr>
<tr>
<td>5-7</td>
<td>User Instructions</td>
</tr>
<tr>
<td>9-10</td>
<td>I/O</td>
</tr>
<tr>
<td>11</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>

4/71
DESCRIPTIVE TITLE  Step-Up and Step-Down Multiple Linear Regression

CALLING NAME  (STPAC) UPREG/DNREG

INSTALLATION NAME  The Pennsylvania State University Computation Center

AUTHOR(S) AND AFFILIATION(S)  Dennis Deaven (programming)
The Pennsylvania State University Computation Center
M.A. Efroymson (flow charts)

LANGUAGE  FORTRAN IV

COMPUTER  IBM System 360/67

PROGRAM AVAILABILITY  Decks and listings presently available

CONTACT  William H. Verity, 104 Computer Building,
The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

FUNCTIONAL ABSTRACT

UPREG/DNREG computes stepwise multiple-regression equations, adding or deleting one variable at a time until no variables remain to add significantly to the equations or until no significant variables remain in the equation. At each step, the following are printed for those variables currently in the equation.

1. Regression coefficients
2. Standard deviation of the regression coefficients
3. Standard regression coefficients
4. Partial correlation coefficients
5. Fraction of explained variance
6. Multiple correlation coefficient
7. Regression-equation intercept
8. Standard deviation of the dependent variable (UPREG only)
9. F value for the variable entering or leaving the equation (UPREG only)

continued
UPREG

In the step-up procedure, intermediate results are used to give statistical information at each step in the calculation. These intermediate answers are also used to control the method of calculation. A number of intermediate-regression equations are obtained, as well as the final multiple-regression equation. These equations are obtained by adding or deleting one variable at a time, giving the following equations.

\[ Y = b'(0) + b'(1)x(1), \]
\[ Y = b''(0) + b''(1)x(1) + b''(2)x(2), \]
\[ Y = b'''(0) + b'''(1)x(1) + b'''(2)x(2) + b'''(3)x(3) \]

The variable added is the one that makes the greatest improvement in goodness of fit. The coefficients represent the best values when the equation is fitted by using the specific variables included in the equation. An important property of this procedure is that a variable may be indicated to be significant in any early stage and thus enter the regression equation. In addition, after several other variables are added to the equation, the initial variable may be indicated to be insignificant. The insignificant variable will be removed from the regression equation before adding an additional variable. Therefore, only significant variables are included in the final multiple-regression equation.

DNREG

The step-down procedure of DNREG is simpler than the step-up procedure of UPREG. If the parsimony option has been chosen, DNREG eliminates one variable at a time, printing the intermediate results at each step. The variable eliminated is the one that contributes least to the overall significance. Note that, once a variable has been eliminated with the procedure, it is never brought back into the equation.

REFERENCES

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute Card is essential. If PPMCR [EIN No. 000 0051(d)] is executed during the same run and Name(s) Card(s) were used, then the Variable-Name(s)-Transfer Card should be used. Note that a Variables Card is not required since PPMCR either passes the number of variables or punches it onto a card to be read by UPREG later on. Some additional required and optional Control Cards are specific to UPREG and DNREG. These are described below.

Dependent-Variable Card (one required)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-9</td>
<td>DEPENDENT</td>
</tr>
<tr>
<td>11-14</td>
<td>xxx</td>
</tr>
</tbody>
</table>

This card designates the variable to be used as the dependent variable. xxx is either the name or the number of one of the variables.

IN Variables Card (optional; one or more)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>IN</td>
</tr>
<tr>
<td>4-</td>
<td>xxx yyy zzz ···</td>
</tr>
</tbody>
</table>

This card selects variables to be forced into the equation no matter what their significance. xxx, yyy, and zzz are either the numbers or names of some of the variables.

OUT Variables Card (optional; one or more)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-3</td>
<td>OUT</td>
</tr>
<tr>
<td>5-</td>
<td>xxx yyy zzz ···</td>
</tr>
</tbody>
</table>

continued
This card selects variables that are never wanted in the equation, no matter how significant any might be. xxx, yyy, and zzz are either the numbers or names of some of the variables.

Tolerance Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-9</td>
<td>TOLERANCE</td>
</tr>
<tr>
<td>11-13</td>
<td>xxx</td>
</tr>
</tbody>
</table>

This card changes the tolerance used in pivoting through the matrix. The default tolerance is 0.07. The correlation matrix is inverted by using a Gauss-Jordan technique and selecting the largest diagonal element at any given stage on which to pivot. If this pivot should become small, it often indicates a degeneracy in the correlation matrix, which indicates that one of the "independent" variables is, in fact, approximately a linear combination of other independent variables. If the tolerance is set too low, the program may develop erroneous results owing to inverting a singular correlation matrix. In any case, the tolerance must be greater than zero.

F-Value Card (optional; UPREG only)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-6</td>
<td>FVALUE</td>
</tr>
<tr>
<td>8-10</td>
<td>xxx</td>
</tr>
</tbody>
</table>

This card selects the critical F value that the program uses to determine whether to add or delete a variable from the equation. The default F value is 2.5.

Parsimony Card (optional; DNREG only)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-9</td>
<td>PARSIMONY</td>
</tr>
</tbody>
</table>

This card indicates to DNREG that the step-down procedure should be used, eliminating one variable at a time from the equation. If the parsimony option is not selected, only one set of results will be printed for each problem—that is, one equation with all variables present except those declared OUT.

Description of Input Data

Data for UPREG and DNREG are a set of means, standard deviations, and correlation coefficients, along with the number of variables.
and number of observations. Normally, the input comes from PPMCR (from a previous problem during the same run using the Input-From PPMCR Card or by using a deck that was punched by PPMCR during some previous job). If the input is supplied by cards, they must be identical in format to those punched by PPMCR.

**Description of Output**

The output from UPREG and DNREG consists of a series of tables, one table being produced each time a variable is added to or deleted from the equation. Each table contains all the information specified in the Functional Abstract.
MULTIPLE REGRESSION ANALYSIS FOR: SNEDECOR EXAMPLE ON PAGE 414

TOLERANCE = .210

TOTAL NUMBER OF VARIABLES = 3
NUMBER OF OBSERVATIONS = 18

F VALUE = 2.50000

DEPENDENT VARIABLE = \( Y_1 \)

START REGRESSION, STANDARD DEVIATION OF THE DEPENDENT VARIABLE: \( \sigma = 26.996 \)
### Step Number 1

**Variable Entering:**
- Variable: X1
- Standard Error of Estimate: 0.205100
- F Value: 0.148170
- Intercept: 0.592590
- Multiple Correlation Coefficient: 0.69340
- Fraction of Explained Variance: 0.48081

**Independent Variable Number and Name**

<table>
<thead>
<tr>
<th>Source of Variation</th>
<th>DF</th>
<th>Sum of Squares</th>
<th>Mean Square</th>
<th>F Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Due to REG</td>
<td>1</td>
<td>5.95702E+03</td>
<td>5.95702E+03</td>
<td>1.48170</td>
</tr>
<tr>
<td>About REG</td>
<td>16</td>
<td>6.43259E+03</td>
<td>4.02037E+02</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>17</td>
<td>1.23896E+04</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Diagonal Elements**

<table>
<thead>
<tr>
<th>Variable Identification</th>
<th>Value</th>
<th>Variable Identification</th>
<th>Value</th>
<th>Variable Identification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 X1</td>
<td>1.000 ENTR</td>
<td>2 X2</td>
<td>0.787</td>
<td>3 Y</td>
<td>0.519 DEP</td>
</tr>
</tbody>
</table>

---

**Standard Deviation and Coefficient for Coefficient**

- Standard Deviation: 0.14344E+01
- Coefficient: 0.47891E+00
- Multiple Correlation Coefficient: 0.69340
- Fraction of Explained Variance: 0.48081
SAMPLE OUTPUT—DNREG

MULTIPLE REGRESSION ANALYSIS FOR: SNEDECOR EXAMPLE ON PAGE 414
PARSIMONY OPTION SELECTED
TOLERANCE = .010
TOTAL NUMBER OF VARIABLES = 3
NUMBER OF OBSERVATIONS = 18

NUMBER OF VARIABLES ELIGIBLE FOR ELIMINATION BY PARSIMONY: 2
THE NUMBER ASSOCIATED WITH THESE VARIABLES AND THEIR NAMES, IF SPECIFIED, ARE LISTED BELOW.
1 (X1), 21 (X2)

DEPENDENT VARIABLE (NAME, IF SPECIFIED, AND ASSOCIATED NUMBER): 31 (Y1)

continued
INDEPENDENT VARIABLE  REGRESSION  STAND.DEVIATION  STAND.REGRESSION  PARTIAL CORRELATION
NUMBER AND NAME  COEFFICIENT  FOR COEFFICIENT  COEFFICIENT  COEFFICIENT
1  X1  0.178977D 01  0.446743D 00  0.673216 00  0.6387
2  X2  0.866442D 01  0.414443D 00  0.043731 00  0.0538

NUMBER OF VARIABLES PRESENTLY ELIMINATED: 0
CANDIDATE FOR THE NEXT ELIMINATION-VARIABLE: 21  X2
F-RATIO OF VARIANCE OF AGGREGATE ELIMINATED AND CANDIDATE VARIABLES TO UNEXPLAINED VARIANCE WITH NONE ELIMINATED: 0.4361D-01
FRACTION OF EXPLAINED VARIANCE: 0.482
MULTIPLE CORRELATION COEFFICIENT: 0.6945
INTERCEPT REG: 0.562510D 02

ANALYSIS OF VARIANCE

<table>
<thead>
<tr>
<th>SOURCE OF VARIATION</th>
<th>SUM OF SQUARES</th>
<th>MEAN SQUARE</th>
<th>F RATIO</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUE TO REG</td>
<td>5.975670D 03</td>
<td>2.987830 03</td>
<td>6.987510 03</td>
</tr>
<tr>
<td>ALONG REG</td>
<td>6.413940D 03</td>
<td>4.275950 02</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>1.238960D 04</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th>INDEPENDENT VARIABLE</th>
<th>REGRESSION COEFFICIENT</th>
<th>STANDARD DEVIATION</th>
<th>STANDARD PARTIAL COEFFICIENT</th>
<th>PARTIAL CORRELATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>0.1843440 D 01</td>
<td>0.47897,2D 00</td>
<td>0.6934031</td>
<td>0.6934</td>
</tr>
</tbody>
</table>

Number of variables presently eliminated: 1

The number associated with the eliminated variables and their names, if specified, are listed below:

- X1

Candidate for the next elimination-variable: 1 (X1)

F-ratio of variance of aggregate eliminated and candidate variables to unexplained variance with none eliminated: 1.6994D 31

Fraction of explained variance: 0.481

Multiple correlation coefficient: 1.6934

Intercept: 8101 0.99299900 02

Analysis of Variance

<table>
<thead>
<tr>
<th>SOURCE OF VARIATION</th>
<th>SUM OF SQUARES</th>
<th>MEAN SQUARES</th>
<th>F RATIO</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUE TO REG</td>
<td>5.95702D 03</td>
<td>5.95702D 03</td>
<td>1.48171D 01</td>
</tr>
<tr>
<td>ABOUT REG</td>
<td>6.43259D 01</td>
<td>4.02587D 02</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>17 1.23848D 04</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Value of determinant = 0.78700 CO

Average absolute deviation from original elements of inverse-inverse matrix elements = 0.20046D-16
COST ESTIMATE
For the job listed on the Sample Input, the total running time on the central processor unit was 6 seconds, at $0.11 per second.

Charge to user = computer time + network overhead
= $0.66 + network overhead

CONTENTS—(STPAC) UPREG/DNREG

pages
1– 2 Identification & Abstract
3– 5 User Instructions
7–12 I/O
13 Cost—Contents
Factor Analysis (or Principal-Components Analysis)

(STPAC) FANAL

The Pennsylvania State University Computation Center

J. Cooley
D. Laird
L. Pryor
J. McConnochie

The Pennsylvania State University Computation Center

FORTRAN IV

IBM System 360/67

Decks and listings presently available

William H. Verity, 104 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

FANAL solves successively for the most dominant factors represented in a (symmetric) correlation matrix. Factors are ranked according to the variance accounted for. The number of factors to be extracted is specified in advance. The program may also be used to find the dominant eigenvalues and eigenvectors of other (symmetric) positive definite matrices.

All computations are done in double-precision floating-point arithmetic. The method used is an iterative one in which a vector converges to the eigenvector. The largest eigenvalue is found first and then a deflation process is used to find the successively smaller eigenvalues. An extrapolation procedure is used to accelerate convergence of the basic iterative method.
USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Description of Input

The input to FANAL consists of the Control Cards described below and the lower half of the symmetric matrix plus optional diagonal elements. Normally, the symmetric matrix is the output from PPMCR [EIN No. 000 0051(d)], another STPAC program, which computes the Pearson product-moment correlation coefficients for a set of variables. The output from PPMCR may be in the form of punched cards received from some previous job or it may be passed directly to FANAL by the Input-From PPMCR option.

The symmetric matrix may also come from another program in STPAC or from a user's program. At present, PHICO [EIN No. 000 0051(k)], SPRHO [EIN No. 000 0051(m)], and PPMCR can pass their results to FANAL. In the case of a user's program, the data must be punched in exactly the same format that PPMCR punches its output.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute Card is required. If the data were punched by the user, a Variables Card is required also.

Factors Card (one or none)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-7</td>
<td>FACTORS</td>
</tr>
<tr>
<td>9-80</td>
<td>n, the number of factors to be extracted from the matrix</td>
</tr>
</tbody>
</table>

If this card is omitted, the program will attempt to extract as many factors as there are variables so long as the variance accounted for is greater than or equal to the minimum variance.

Minimum-Variance Card (one or none)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-16</td>
<td>MINIMUM VARIANCE</td>
</tr>
<tr>
<td>18-80</td>
<td>x, any floating-point number, with the decimal point punched</td>
</tr>
</tbody>
</table>

continued
x indicates the eigenvalue at which factor extraction will be terminated. If this card is not supplied, the default minimum variance will be zero.

Correlation Card (one or none)

**Columns**  **Contents**
1-18  CORRELATION MATRIX

This card is a request that the input matrix be printed out.

Residual Card (one or none)

**Columns**  **Contents**
1-15  RESIDUAL MATRIX

This card is a request that the residual matrix be printed out prior to termination of the program.

Diagonal Card (one or none)

**Columns**  **Contents**
1  8  DIAGONAL
10-80  d, a floating-point number, with the decimal point punched, followed by an optional E or D field

d is the value to be placed into each diagonal element of the matrix. An alternative version permits the entry of unique diagonal elements.

**Columns**  **Contents**
1-14  DIAGONAL CARDS

This card indicates that the diagonal elements will be read from Data Cards following the Begin-Data Card. They must be punched according to a specific format.

**Columns**  **Contents**
1  Blank
2  D
3  Blank
4-6  Card sequence number
7-18  1st diagonal element on this card
67-78  6th diagonal element on this card

continued
Thus, card 1 would contain the diagonal elements (1,1) through (6,6) and card 5, for example, would contain the diagonal elements (25,25) through (30,30). Note that this format is identical to that used by PPMCR to punch its means and standard deviations, except for the letter D, which designates these cards as diagonal cards.

**Description of Output**

For each factor, the output includes the variance accounted for (the eigenvalue $\lambda_i$), the eigenvector $U_i$, and the vector of factor loadings $f_i$, where $f_i = U_i/\lambda_i$.

At the user's option, the input matrix can be printed out and, at the completion of the factor extraction, the residual matrix can be printed out. Also, a punched deck can be requested to be entered at a later time to VARMX [EIN No. 000 0051(j)].
SAMPLE INPUT

EXECUTE PPNC
TITLE SAMPLE INPUT
VARIABLES X Y
VARIABLE NAMING X1, Y1
FORMAT 4,6,7,1,1,1
BEGIN DATA
0064344
0064761
0064261
0064244
0172677
0044641
1033114
1021801
1043241
2014444
2014777
2014444
1102641
2294140
2294140
2294140
2294140
2294140
2294140
END DATA
EXECUTE STOP
EXECUTE END
**SAMPLE OUTPUT**

PRINCIPAL COMPONENTS ANALYSIS - SEPTEMBER 1964 VERSION - THE PENNSYLVANIA STATE UNIVERSITY COMPUTATION CENTER.

**PROBLEM NAME:** SNEDECOR EXAMPLE ON PAGE 414

This problem has 3 variables and 3 factors.

<table>
<thead>
<tr>
<th>FACTOR NUMBER</th>
<th>TEST NUMBER</th>
<th>FACTOR</th>
<th>VARIANCE</th>
<th>EIGENVECTOR</th>
<th>ITERATIONS REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X1</td>
<td>1</td>
<td>2.02068</td>
<td>0.63032</td>
<td>0.47324</td>
</tr>
<tr>
<td></td>
<td>X2</td>
<td>2</td>
<td>0.70114</td>
<td>0.49324</td>
<td>0.56991</td>
</tr>
<tr>
<td></td>
<td>Y</td>
<td>3</td>
<td>0.85220</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>X1</td>
<td>1</td>
<td>0.70716</td>
<td>0.22495</td>
<td>-0.70775</td>
</tr>
<tr>
<td></td>
<td>X2</td>
<td>2</td>
<td>0.47772</td>
<td>0.85515</td>
<td>-0.15952</td>
</tr>
<tr>
<td></td>
<td>Y</td>
<td>3</td>
<td>0.38653</td>
<td>0.46702</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>X1</td>
<td>1</td>
<td>0.30311</td>
<td>0.74302</td>
<td>-0.17997</td>
</tr>
<tr>
<td></td>
<td>X2</td>
<td>2</td>
<td>-0.38643</td>
<td>-0.17997</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Y</td>
<td>3</td>
<td>-0.35263</td>
<td>-0.64999</td>
<td></td>
</tr>
</tbody>
</table>
COST ESTIMATE

For the job listed on the Sample Input, the total running time on the central processor unit was 6 seconds, at $0.11 per second.

Charge to user = computer time + network overhead

= $0.66 + network overhead

CONTENTS—(STPAC) FANAL

pages

1 Identification & Abstract
3–5 User Instructions
7–8 I/O
9 Cost—Contents

4/71
**FUNCTIONAL ABSTRACT**

VARMX can perform a number of orthogonal rotations on an arbitrary matrix of factor loadings, using the normal varimax criterion. The result is a unique (within tolerance limits) matrix of factor loadings.

All computations are done in double-precision floating-point arithmetic. First, the original factor loadings as produced by FANAL [EIN No. 000 0051(i)] are normalized by dividing the loadings in each row by the square root of the communality (sum of squares) for that row. The rotation is performed with a criterion of .005 and the resulting loadings are denormalized to give the rotated matrix.

**REFERENCES**

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Description of Input

The input to VARMX consists of the Control Cards described below and the matrix of factor loadings to be rotated. The factor matrix is the output from FANAL [EIN No. 000 0051(i)], either passed directly to VARMX by using the Input-From FANAL option or in the form of a card deck punched by FANAL during some previous job.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute Card is required and either an Input-From FANAL Card or a Begin-Data/End-Data sequence with a deck punched by FANAL. There are some additional Control Cards specific to VARMX. They are discussed below.

Rotate Card(s) (none, one, or more)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-6</td>
<td>ROTATE</td>
</tr>
<tr>
<td>8-80</td>
<td>n₁...nᵦ</td>
</tr>
</tbody>
</table>

This general form of the Rotate Card requests that k distinct rotations be performed, first rotating the first n₁ factors, then the first n₂ factors, and finally the first nᵦ factors. If this card is omitted, the entire matrix of factor loadings will be rotated.

NOTE: When VARMX is completed, the output from FANAL is no longer available. It is destroyed after the last rotation request is completed.

Print Card(s) (one or more)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>PRINT</td>
</tr>
</tbody>
</table>

This card requests a printout of the input matrix.

continued
This card requests a punched deck of the matrix of rotated factor loadings. For each rotation, a set of cards will be punched for each variable, with each set of cards containing the corresponding rotated factor loadings for that variable. The format of the punched cards is ('VAR', I3, 'CD', I2, 7F10.5). The first integer field is the variable number, the second is a card sequence number for this variable, and the seven floating-point fields are for the rotated loadings. This deck can then be submitted to PPMCR [EIN No. 000 0051(d)] to correlate the rotated factors. Actually, a Format Card is punched along with some other Control Cards so that about all that the deck needs to submit it to PPMCR is an Execute PPMCR Card.

Description of Output

For each rotation, the output includes the communalities and the rotated matrix of factor loadings. At the user's option, the original input matrix can be printed out and a deck of cards containing the rotated matrix can be obtained for possible submission to PPMCR at a later date.
SAMPLE INPUT

EXECUTE PPNCR

TITLE SAMPLE INPUT ON PAGE 414

VARIABLES 3

VARIABLE NAMES X1,Y1,Z1

FORMAT (3, 7F12.7)

BEGIN DATA

01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70

END DATA

EXECUTE SIGDP

INPUT FROM PPNCR

LEVEL .1 TEN

EXECUTE MODEG

INPUT FROM PPNCR

PRINTLEVEL Y

EXECUTE MEMORY

INPUT FROM PPNCR

PRINTLEVEL Y

EXECUTE CANON

INPUT FROM PPNCR

LEFT Y

EXECUTE X1, X2

EXECUTE FANAL

INPUT FROM PPNCR

FACTORS 3

EXECUTE VARMS

INPUT FROM FANAL

STOP

------0051(d)

------0051(e)

------0051(h)

------0051(g)

------0051(i)

------0051(j)
VARIMAX ROTATION
SNEDECOR EXAMPLE ON PAGE 414

THE INPUT MATRIX FROM FACTOR ANALYSIS CONTAINS 3 VARIABLES AND 3 FACTORS.

**ROTATION OF FIRST 3 FACTORS**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Communalities</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>1.00000</td>
</tr>
<tr>
<td>X2</td>
<td>1.00000</td>
</tr>
<tr>
<td>Y</td>
<td>1.00000</td>
</tr>
</tbody>
</table>

**THE ROTATED MATRIX OF FACTOR LOADINGS**

<table>
<thead>
<tr>
<th></th>
<th>F1</th>
<th>F2</th>
<th>F3</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>0.37412</td>
<td>-0.23774</td>
<td>0.89639</td>
</tr>
<tr>
<td>X2</td>
<td>0.14472</td>
<td>-0.96959</td>
<td>0.19737</td>
</tr>
<tr>
<td>Y</td>
<td>0.92506</td>
<td>-0.15712</td>
<td>0.34580</td>
</tr>
</tbody>
</table>
COST ESTIMATE
For the job listed on the Sample Input, the total running time on central processor unit was 6 seconds, at $0.11 per second.
Charge to user = computer time + network overhead
   = $0.66 + network overhead

CONTENTS—(STPAC) VARMX

<table>
<thead>
<tr>
<th>pages</th>
<th>Identification &amp; Abstract</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3-4</td>
<td>User Instructions</td>
</tr>
<tr>
<td>5-6</td>
<td>I/O</td>
</tr>
<tr>
<td>7</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>
**FUNCTIONAL ABSTRACT**

PHICO processes a maximum of 105 variables and computes phi coefficients for all possible pairs of variables. Data are dichotomous and are coded 0 and 1 or 1 and 2. The equation used is

\[ \phi = \frac{(ad-bc)}{\sqrt{(a+b)(c+d)(a+c)(b+d)}} \]

**REFERENCES**

EDUCOM
EDUCATIONAL INFORMATION NETWORK

000 0051(k)

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various STPAC Control Cards discussed in the STPAC writeup, the Execute, Variables, and Format Cards are essential for PHICO. Begin- and End-Data Cards are also necessary if the data have not already been read for a previous STPAC program. All other STPAC Control Cards are optional. There are some additional required and optional Control Cards specific to PHICO. These are described below.

Code Card (one required)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>CODE</td>
</tr>
</tbody>
</table>
| 6       | A: data punched 0 and 1  
|         | B: data punched 1 and 2 |

Punch Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>PUNCH</td>
</tr>
</tbody>
</table>

If this card is included, phi coefficients will be punched on cards in addition to being printed.

Description of Input Data

Data must be punched according to the format given on the Format Card. On each card should be punched a case and sequence number (only if required, as described in the STPAC writeup) and the data points in I format. The "number" of a variable is determined by the order in which it is read, which is determined by the format supplied.

Description of Output

Phi coefficients are printed in a triangular matrix. All variables are numbered. Rows and columns will also be named according to the Variable-Names Card, if included. Card output is also produced when requested.
SAMPLE INPUT

EXECUTE PUNCH
TITLE TEST OF TS AND PS
VARIABLES FS 1S
BEGIN DATA
171111122221111
212111122221111
121222211121221
1712111122221111
121211122221111
1712111122221111
1212111122221111
END DATA
STOP
### TEST OF IS AND ZS

**PHI COEFFICIENT SYMMETRIC MATRIX**

NUMBER OF CASES 8

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1.00</td>
<td>-1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.40</td>
<td>0.40</td>
<td>0.24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.14</td>
<td>0.14</td>
<td>0.14</td>
<td>0.29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.25</td>
<td>0.37</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.21</td>
<td>0.21</td>
<td>0.21</td>
<td>0.21</td>
<td>-0.21</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
</tr>
<tr>
<td>8</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
</tr>
<tr>
<td>9</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
</tr>
<tr>
<td>10</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
<td>-0.07</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
</tr>
<tr>
<td>11</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
</tr>
<tr>
<td>12</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
<td>0.48</td>
<td>-0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>-0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
</tr>
<tr>
<td>13</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
<td>0.37</td>
</tr>
<tr>
<td>14</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>-0.29</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
</tr>
<tr>
<td>15</td>
<td>0.14</td>
<td>0.14</td>
<td>0.14</td>
<td>0.14</td>
<td>-0.14</td>
<td>-0.14</td>
<td>-0.14</td>
<td>-0.14</td>
<td>-0.14</td>
<td>-0.14</td>
<td>-0.14</td>
<td>-0.14</td>
</tr>
</tbody>
</table>

---

EDUCATIONAL INFORMATION NETWORK
COST ESTIMATE
For the job listed on the Sample Input, the total running time was 4 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
               = $0.44 + network overhead

CONTENTS—STPAC

pages                                    
  1  Identification & Abstract            
  3  User Instructions                    
  5–6 I/O                                 
  7  Cost—Contents
KENDALL RAND CORRELATION COEFFICIENTS (tau) (STPAC) KETAU

The Pennsylvania State University Computation Center

Nancy C. Daubert
The Pennsylvania State University Computation Center

FORTRAN IV

IBM System 360/67

Decks and listings presently available

William H. Verity, 104 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

FUNCTIONAL ABSTRACT

KETAU computes Kendall rank-correlation coefficients (tau), the values of z scores corresponding to them, and the probabilities and levels of significance of the z scores, for all possible pairs of variables. A correction is made for ties. The correlation coefficients will also be punched on cards, at the option of the user.

REFERENCES

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various STPAC Control Cards discussed in the STPAC writeup, the Execute, Variables, and Format Cards are essential for KETAU. Begin- and End-Data Cards are also necessary if the data have not been already read for a previous STPAC program. All other STPAC Control Cards are optional. There is one additional optional Control Card specific to KETAU. This is described below.

Punch Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–5</td>
<td>Punch</td>
</tr>
</tbody>
</table>

This card should be included if the user wishes to have the values of tau punched, in addition to being printed.

Description of Input Data

Data for all variables should be punched together. The number of a variable is determined by the order in which it is read, which is determined by the format supplied. Thus, the first variable read will be variable 1, the second will be variable 2, and so on. There is no standard format for raw or ranked data, but the data must be in F or E format. Sequence or case numbers, if needed, must be in I format.

Description of Output

The printed output for a KETAU problem consists of four triangular matrices. All are headed by the title that the user supplies. All variables are numbered and both rows and columns will be named according to the Variable-Names Card, if it is included. The first matrix consists of the values of tau for each possible pair of variables. The second is a table of z scores associated with the corresponding coefficients. The third give the probabilities of the z scores and the fourth gives their level of significance. If the total number of observations is less than 10, z scores should be interpreted with caution. However, special tables of the critical values of tau in this range are available.
### SAMPLE INPUT

```plaintext
EVERYONE READ
END DATA
END DATA
3 2 10
2 5 10
1 1 10
2 1 10
1 2 10
1 0 10
1 3 10
9 4 10
1 2 10
END DATA
END DATA
STOP
```
KENDALL RANK CORRELATION COEFFICIENT

NUMBER OF OBSERVATIONS IS 12

KENDALL'S RANK CORRELATION COEFFICIENT

| Striving | Authorit- | Authorit-
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>0.6667</td>
<td>0.3722</td>
</tr>
</tbody>
</table>

Corresponding Z scores

| Striving | Authorit- | Authorit-
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>3.0172</td>
<td>1.6845</td>
</tr>
</tbody>
</table>

Corresponding probabilities of Z scores (p > 0.05)

Levels of significance

<table>
<thead>
<tr>
<th>Levels</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>blank</td>
<td>p &lt; 0.001</td>
</tr>
<tr>
<td>*</td>
<td>0.001 &lt; p &lt; 0.01</td>
</tr>
<tr>
<td>**</td>
<td>0.01 &lt; p &lt; 0.05</td>
</tr>
<tr>
<td>***</td>
<td>0.05 &lt; p &lt; 0.10</td>
</tr>
<tr>
<td>****</td>
<td>p &lt; 0.05</td>
</tr>
</tbody>
</table>
COST ESTIMATE

For the job listed on the Sample Input, the total running time on the central processor unit was 4 seconds at $0.11 per second.

Charge to user = computer time + network overhead
= $0.44 + network overhead

CONTENTS—(STPAC) KETAU

pages
1 Identification & Abstract
3 User Instructions
5–6 I/O
7 Cost—Contents
DESCRIPTIVE TITLE: Spearman Rank Correlation Coefficients (rho)

CALLING NAME: (STPAC) SPRHO

INSTALLATION NAME: The Pennsylvania State University Computation Center

AUTHOR(S) AND AFFILIATION(S): Nancy C. Daubert, The Pennsylvania State University Computation Center

LANGUAGE: FORTRAN IV

COMPUTER: IBM System 360/67

PROGRAM AVAILABILITY: Decks and listings presently available

CONTACT: William H. Verity, 104 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

FUNCTIONAL ABSTRACT:
SPRHO computes Spearman rank-order correlation coefficients (rho) and the values of t corresponding to them for all possible pairs of variables. A correction is made for ties. The correlation coefficients will also be punched on cards at the option of the user.

REFERENCES:
USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various STPAC Control Cards discussed in the STPAC writeup, the Execute, Variables, and Format Cards are essential for SPRHO. Begin- and End-Data Cards are also necessary if the data have not been already read for a previous STPAC program. All other STPAC Control Cards are optional. There is one additional optional Control Card specific to SPRHO. It is described below.

Punch Card (optional)

<table>
<thead>
<tr>
<th>Column Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1- 5</td>
</tr>
<tr>
<td>PUNCH</td>
</tr>
</tbody>
</table>

This card should be included if the user wishes to have the correlation coefficients punched in addition to being printed.

Description of Input Data

Data for all variables within one observation should be punched together. The number of a variable is determined by the order in which it is read, which is determined by the format supplied. Thus, the first variable read will be variable 1, the second will be variable 2, etc. There is no standard format, but the data must be in F or E format. Sequence and case numbers, if needed, must be in I format.

Description of Output

The printed output for a SPRHO problem consists of two triangular matrices. Both are headed by the title that the user supplies. All variables are numbered and both rows and columns will also be named according to the Variable-Name(s) Card, if it is included. The first matrix consists of the Spearman rank-order correlation coefficients for all possible pairs of variables. The second is a table of the values of t associated with the corresponding values of rho. Degrees of freedom are given as part of the heading. If the number of observations is less than 10, these values should be interpreted with caution. However, special tables of critical values of rho in this range are available.
<table>
<thead>
<tr>
<th></th>
<th>ONE</th>
<th>TWO</th>
<th>THREE</th>
<th>FOUR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.865</td>
<td>0.752</td>
<td>0.726</td>
<td>0.728</td>
</tr>
<tr>
<td>2</td>
<td>0.906</td>
<td>0.880</td>
<td>0.950</td>
<td>0.991</td>
</tr>
<tr>
<td>3</td>
<td>-0.127</td>
<td>-0.167</td>
<td>-0.203</td>
<td>-0.150</td>
</tr>
<tr>
<td>4</td>
<td>-0.007</td>
<td>-0.007</td>
<td>-0.007</td>
<td>-0.007</td>
</tr>
<tr>
<td>5</td>
<td>0.098</td>
<td>0.136</td>
<td>0.207</td>
<td>0.237</td>
</tr>
<tr>
<td>6</td>
<td>0.976</td>
<td>0.91</td>
<td>0.884</td>
<td>0.773</td>
</tr>
<tr>
<td>7</td>
<td>-0.046</td>
<td>-0.018</td>
<td>-0.198</td>
<td>-0.183</td>
</tr>
<tr>
<td>8</td>
<td>-0.023</td>
<td>-0.040</td>
<td>-0.207</td>
<td>-0.255</td>
</tr>
<tr>
<td>9</td>
<td>0.091</td>
<td>0.056</td>
<td>0.010</td>
<td>-0.381</td>
</tr>
<tr>
<td>10</td>
<td>-0.214</td>
<td>-0.183</td>
<td>-0.209</td>
<td>-0.298</td>
</tr>
<tr>
<td>11</td>
<td>-0.281</td>
<td>-0.220</td>
<td>-0.215</td>
<td>-0.384</td>
</tr>
<tr>
<td>12</td>
<td>-0.351</td>
<td>-0.303</td>
<td>-0.263</td>
<td>-0.404</td>
</tr>
<tr>
<td>13</td>
<td>-0.317</td>
<td>-0.252</td>
<td>-0.303</td>
<td>-0.298</td>
</tr>
<tr>
<td>14</td>
<td>-0.284</td>
<td>-0.220</td>
<td>-0.198</td>
<td>-0.183</td>
</tr>
<tr>
<td>15</td>
<td>-0.220</td>
<td>-0.167</td>
<td>-0.136</td>
<td>-0.088</td>
</tr>
<tr>
<td>16</td>
<td>-0.114</td>
<td>-0.065</td>
<td>-0.020</td>
<td>-0.007</td>
</tr>
<tr>
<td>17</td>
<td>0.104</td>
<td>0.075</td>
<td>0.038</td>
<td>0.007</td>
</tr>
<tr>
<td>18</td>
<td>0.061</td>
<td>0.023</td>
<td>0.007</td>
<td>0.007</td>
</tr>
<tr>
<td>19</td>
<td>0.034</td>
<td>0.010</td>
<td>0.004</td>
<td>0.001</td>
</tr>
<tr>
<td>20</td>
<td>0.004</td>
<td>0.002</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

**SPEARMAN RANK CORRELATION COEFFICIENT**

**NUMBER OF OBSERVATIONS IS 30**
## Demonstration of Sφh0

### Corresponding Values of Student’s Degrees of Freedom

<table>
<thead>
<tr>
<th>Two</th>
<th>Three</th>
<th>Four</th>
<th>Five</th>
<th>Six</th>
<th>Seven</th>
<th>Eight</th>
<th>Nine</th>
<th>Ten</th>
<th>Eleven</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>5.648</td>
<td>11.249</td>
<td>16.095</td>
<td>20.095</td>
<td>17.661</td>
<td>34.924</td>
<td>13.819</td>
<td>10.897</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>5.849</td>
<td>6.357</td>
<td>5.187</td>
<td>5.506</td>
<td>5.313</td>
<td>5.923</td>
<td>34.924</td>
<td>13.819</td>
<td>10.897</td>
</tr>
<tr>
<td>14</td>
<td>5.001</td>
<td>5.438</td>
<td>5.836</td>
<td>5.506</td>
<td>5.313</td>
<td>5.923</td>
<td>34.924</td>
<td>13.819</td>
<td>10.897</td>
</tr>
<tr>
<td>15</td>
<td>6.865</td>
<td>5.173</td>
<td>5.370</td>
<td>4.960</td>
<td>4.664</td>
<td>5.182</td>
<td>34.924</td>
<td>13.819</td>
<td>10.897</td>
</tr>
<tr>
<td>16</td>
<td>5.208</td>
<td>5.598</td>
<td>6.325</td>
<td>5.144</td>
<td>4.045</td>
<td>5.812</td>
<td>34.924</td>
<td>13.819</td>
<td>10.897</td>
</tr>
<tr>
<td>17</td>
<td>4.327</td>
<td>6.291</td>
<td>7.831</td>
<td>5.780</td>
<td>5.732</td>
<td>5.843</td>
<td>34.924</td>
<td>13.819</td>
<td>10.897</td>
</tr>
<tr>
<td>18</td>
<td>4.227</td>
<td>6.291</td>
<td>7.831</td>
<td>5.780</td>
<td>5.732</td>
<td>5.843</td>
<td>34.924</td>
<td>13.819</td>
<td>10.897</td>
</tr>
<tr>
<td>19</td>
<td>1.446</td>
<td>1.323</td>
<td>1.303</td>
<td>1.027</td>
<td>1.265</td>
<td>1.397</td>
<td>1.680</td>
<td>1.019</td>
<td>0.336</td>
</tr>
<tr>
<td>20</td>
<td>0.122</td>
<td>0.570</td>
<td>0.033</td>
<td>0.107</td>
<td>0.345</td>
<td>0.353</td>
<td>0.521</td>
<td>1.016</td>
<td>0.161</td>
</tr>
<tr>
<td>21</td>
<td>-0.80</td>
<td>-0.375</td>
<td>-0.404</td>
<td>-0.790</td>
<td>-0.525</td>
<td>-0.526</td>
<td>-0.408</td>
<td>0.033</td>
<td>-0.693</td>
</tr>
</tbody>
</table>

### Number of Observations

Number of Observations is 30.
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 5 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
               = $0.55 + network overhead

CONTENT—(STPAC) SPRHO

pages   Identification & Abstract
         User Instructions
         I/O
         Cost—Contents

4/71

9 288
DESCRIPTIVE TITLE
Mann-Whitney U Test

CALLING NAME
(STPAC) MNWHT

INSTALLATION NAME
The Pennsylvania State University Computation Center

AUTHOR(S) AND AFFILIATION(S)
Nancy C. Daubert
The Pennsylvania State University Computation Center

LANGUAGE
FORTRAN IV

COMPUTER
IBM System 360/67

PROGRAM AVAILABILITY
Decks and listings presently available

CONTACT
William H. Verity, 104 Computer Building,
The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

FUNCTIONAL ABSTRACT
MNWHT computes results for the Mann-Whitney U test and gives corresponding z scores, probabilities of the z scores, and levels of significance. U's are calculated for all possible pairs of variables. Observations per variable may be unequal. A correction is made for ties. U's will be punched on cards at the option of the user.

REFERENCES
USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute, Variables, and Format Cards are essential for MNWHT. Begin- and End-Data Cards also are necessary if the data have not been already read for a previous STPAC program. All other STPAC Control Cards are optional. One additional optional Control Card is specific to MNWHT and is described below.

Punch Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>PUNCH</td>
</tr>
</tbody>
</table>

This card should be included only if the user wishes to have the values of U punched on cards in addition to being printed.

Description of Input Data

Each Data Card should have a variable-identification number (in integer format), followed by a data point (in E or F format). Variable-identification numbers need not be successive, but the program's number of a variable is determined by the order in which it is read. All Data Cards for one variable should be together.

Description of Output

The printed output for a MNWHT problem consists of four triangular matrices. All tables are headed by a user-supplied title. All variables are numbered, and both rows and columns will be named according to the Variable-Names Card, if it is included. Variable identification in each table also includes the number of observations for each variable.

The first table consists of the Mann–Whitney U's for all sets of observations. The second table gives corresponding z scores. A z score in this case should be interpreted cautiously if the number of observations for both of the variables involved is less than 20. Special tables are available to determine the associated continued
probability in that range. The third table gives the probability associated with each z score.

The fourth table gives the level of significance associated with each of the probabilities.
SAMPLE INPUT

MADDY MADDY
TITLE ANNOTATION TEST FOR
VARIABLES: A
FORMAT (11,2F7.0)
READ DATA
1 A
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2
1 2

EDUCATIONAL INFORMATION NETWORK
000 0051(n)

SAMPLE OUTPUT

MANN-WHITNEY TEST ONE

MANN - WHITNEY U TEST

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>3</td>
<td>23</td>
<td>5</td>
<td>16</td>
<td>4</td>
</tr>
</tbody>
</table>

- 2 N= 23 29.900
- 3 N= 5  0.0  0.0
- 4 N= 16 10.900 44.300 0.0
- 5 N= 4  3.000 20.500 0.0  30.200
- 6 N= 4  3.0  0.0  9.000 0.3  0.0

MANN-WHITNEY TEST ONE

CORRESPONDING Z SCORES

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>3</td>
<td>23</td>
<td>5</td>
<td>16</td>
<td>4</td>
</tr>
</tbody>
</table>

- 2 N= 23  -0.434
- 3 N= 5   -2.231 -1.462
- 4 N= 16  -1.332 -1.451 -3.338
- 5 N= 4   -1.061 -1.792 -2.699 -0.192
- 6 N= 4   -2.121 -3.133 -0.245 -3.061 -2.309

MANN-WHITNEY TEST ONE

CORRESPONDING PROBABILITIES OF Z SCORES (P(X GREATER THAN OR EQUAL TO Z))

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>3</td>
<td>23</td>
<td>5</td>
<td>16</td>
<td>4</td>
</tr>
</tbody>
</table>

- 2 N= 23  0.361
- 3 N= 5   0.313 0.000
- 4 N= 16  0.063 0.200 0.000 0.424
- 5 N= 4   0.042 0.040 0.007 0.434
- 6 N= 4   0.017 0.001 0.403 0.001 0.010

MANN-WHITNEY TEST ONE

LEVELS OF SIGNIFICANCE

BLANK IF 0.050 < P
* IF 0.025 < P < 0.050
** IF 0.010 < P < 0.025
*** IF 0.001 < P < 0.010
**** IF P < 0.001

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>3</td>
<td>23</td>
<td>5</td>
<td>16</td>
<td>4</td>
</tr>
</tbody>
</table>

- 2 N= 6  
- 3 N= 5  ** ****
- 4 N= 16 ****
- 5 N= 4  ** ****
- 6 N= 4  ** ****
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 4 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
               = $0.44 + network overhead

CONTENTS—(STPAC) MNWHT

pages  identification & abstract  1
       user instructions 3-4
       i/o 5-6
       cost—contents 7
DESCRIPTIVE TITLE  Kruskal–Wallis One-Way Analysis of Variance

CALLING NAME  (STPAC) KRWAL

INSTALLATION NAME  The Pennsylvania State University Computation Center

AUTHOR(S) AND AFFILIATION(S)  Nancy C. Daubert
The Pennsylvania State University Computation Center

LANGUAGE  FORTRAN IV

COMPUTER  IBM System 360/67

PROGRAM AVAILABILITY  Decks and listings presently available

CONTACT  William H. Verity, 104 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

FUNCTIONAL ABSTRACT

KRWAL does a Kruskal–Wallis one-way analysis of variance. The levels need not have equal numbers of observations in each. A correction is made for ties.

REFERENCES

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed by using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various STPAC Control Cards discussed in the STPAC writeup, the Execute and Format cards are essential for KRWAL. Begin-Data and End-Data Cards also are necessary if the data have not been already read for a previous STPAC program. All other STPAC Control Cards are optional. No additional Control Cards are necessary for KRWAL.

Description of Input Data

Each Data Card must contain a level (sample) number in integer format and a data value in E or F format. Level numbers may be any integer values. They need not be consecutive but must be distinct. All Data Cards for a single level (sample) must be together. There is no standard format for this program.

Description of Output

Output consists of degrees of freedom, the value of H, and the chi-square probability of H. If there are ties in the data, the number of groups of ties, the total number of ties, and the correction factor also will be printed. If the number of observations per level (sample) is less than five, caution should be exercised in the interpretation of the chi-square probability. Tables are available for probabilities of H in this lower range.
## SAMPLE INPUT

**FURTHER ARMAL**

**TITLE SECOND TEST OF KRUSKAL WALLIS ANOVA-FIRST DATA RANKED**

**VARIABLES a**

**FORMAT (I3,F3.1)**

**BEGIN DATA**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>84</td>
<td>1275</td>
<td>1475</td>
<td>141</td>
<td>154</td>
</tr>
<tr>
<td>2</td>
<td>154</td>
<td>1475</td>
<td>1275</td>
<td>141</td>
<td>154</td>
</tr>
<tr>
<td>3</td>
<td>1475</td>
<td>1275</td>
<td>141</td>
<td>154</td>
<td>1475</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**END DATA**

R 4

**FIND DATA**

**STOP**
SAMPLE OUTPUT

KRUSKAL-WALLIS ONE WAY ANALYSIS OF VARIANCE BY RANKS
SECOND TEST OF KRUSKAL WALLIS ANOVA-FIRST DATA RANKED
7 DEGREES OF FREEDOM
H = 18.565
13 GROUPS OF TIES, 56 TOTAL TIES. CORRECTION FACTOR IS 0.0244316
CHI SQUARE PROBABILITY = 0.007644
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 4 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
= $0.44 + network overhead

CONTENTS—(STPAC) KRWAL

pages
1 Identification & Abstract
3 User Instructions
5–6 I/O
7 Cost—Contents
Analysis-of-Variance Method of Unweighted Means

(STPAC) ANOVUM

The Pennsylvania State University Computation Center

Nancy C. Daubert
The Pennsylvania State University Computation Center

FORTRAN IV

IBM System 360/67

Decks and listings presently available

William H. Verity, 104 Computer Building, The Pennsylvania State University, University Park, Pa. 16802
Tel.: (814) 865-9527

ANOVUM is designed to handle a factorial analysis of variance with either equal or unequal numbers of observations in subclasses. The method of unweighted means is based on the assumption that the population from which the sample is drawn has proportional or equal subclass numbers. It can be relied upon only if the subclass numbers are approximately equal and presumably represent a population with equal numbers. The analysis is performed upon the means for each cell, and each sum of squares is multiplied by the harmonic mean of the number of observations per cell. Up to 150 variables (separate analyses) will be permitted with data code = 1. Only 1 variable is acceptable with data code = 0.

A Bartlett's test for homogeneity of variance is performed on the raw data. If the chi square computed in this test is significant at the 0.05 level, the conclusions of the analysis of variance should be interpreted cautiously. If cell frequencies are greater than 1, and if they are equal, a Scheffe's test will be performed on main effects.

REFERENCES

USER INSTRUCTIONS

This Program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute and Format Cards are essential for ANOVUM. Begin- and End-Data Cards are also necessary if the data have not already been read for a previous STPAC program. All other STPAC Control Cards are optional. Some additional Control Cards are required and some optional Control Cards are specific to ANOVUM (see below).

Treatments Card (required)

Columns Contents
1–10 TREATMENTS
12 N, the number of treatments
14–19 LEVELS
21–80 \( L_1, L_2, \ldots, L_N; \ L_i \) is the number of levels for the \( i \)th treatment

Data-Code Card (required)

Columns Contents
1–9 DATA CODE
11 1: there is one data point per variable per card, preceded by cell identification
0: all data for a single cell are punched on a single card or card sequence. If more than one Data Card per cell is required (i.e., if cell identification and data use more than 80 columns), a sequence number must be included on each card, following the cell identification. See Data Cards, below.

Equal or Unequal Card (required only if data code is 0)

Columns Contents
1–80 EQUAL n SEQUENCED m PERCARD
or

continued
This card indicates whether the cell frequencies are equal or unequal. \( n \) must be the number of observations per cell if cells are equal or the maximum number of observations per cell if cells are unequal.

The next three fields may be omitted if only one Data Card per cell is necessary. If, however, more than one Data Card per cell is necessary, the last three fields of this card must be included. \( m \), in this case, is the number of data points per card.

**Punch Card (optional)**

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>PUNCH</td>
</tr>
</tbody>
</table>

This card should be included only if the user desires punched-card output of means.

**Cell Summary Card (optional)**

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-12</td>
<td>CELL SUMMARY</td>
</tr>
</tbody>
</table>

This card calls for a summary table of cell frequencies, means, variance, and degrees of freedom to be printed.

**Treatment Summary Card (optional)**

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-17</td>
<td>TREATMENT SUMMARY</td>
</tr>
</tbody>
</table>

This card calls for a summary table of totals, number of observations, means, and sums of squares for each treatment group and combination to be printed.

**Missing-Data Card (optional)**

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-12</td>
<td>MISSING DATA</td>
</tr>
<tr>
<td>14-21</td>
<td>( x ), any number up to eight digits, including a decimal, which will be read as a code for missing data</td>
</tr>
</tbody>
</table>

Include this card only if some data are missing. If this card is omitted, it will be assumed that all fields contain data. (blanks continued)
will be read as zero). If this card is included with Cols. 14–21 blank, zeros and blanks in the Data Cards will be treated as missing data. If Cols. 14–21 are punched, the number x in a data field will be treated as a case of missing data.

Scheffe Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–7</td>
<td>000 0051(p)</td>
</tr>
</tbody>
</table>

Scheffe Card (optional)

Columns Contents

1–7 SCHEFFE

9–80 Desired level of alpha for the Scheffe test. (The default level of alpha is 0.05.)

No standard format is provided for this program; format must be supplied by the user as specified in the requirements for STPAC.

Data Cards

Data Cards are punched with code numbers signifying level for each treatment, starting at the lefthand side of the card (cell identification). These must be in integer format. The first \( I_1 \) indicates level for treatment 1; the second \( I_2 \) indicates level for treatment 2; etc. If any of the \( I_1, \ldots, I_N \) are blank or zero, the card will be ignored by the program. If any of the \( I_1, \ldots, I_N \) are greater than the maximum number of levels specified on the Treatments Card, the analysis will not be performed.

If the data code is 1, a single data point for each of the variables specified (in F, D, or E format) follows the cell identification. Data Cards with this data code may be in any order.

If the data code is 0, and cell frequencies are unequal, the number of data points in that particular cell (in I format) must immediately follow the cell identification. If a sequence number is also required, it (also in I format) follows the number of data points. In the case where more than one card per cell is needed under data code = 0, cell identification, number of data points, and sequence number must be the first items of information on every card. Data fields follow the preceding fields. All cards for a single cell must be together.

Data Code = 1 FORM: \( I_1, \ldots, I_N X_1, \ldots, X_K \)

where \( I_1, \ldots, I_N \) are integer cell-identification numbers, identifying one level for each of the \( N \) treatments; \( X_1, \ldots, X_K \) are data points, one for each of \( K \) specified variables.

continued
Data Code = 0

FORM 1: \[ I_1, \ldots, I_N \ X_1, \ldots, X_n \]
WHEN: cell frequencies are equal
all data for a cell on single card

\[ I_1, \ldots, I_N \] are integer cell-identification numbers, identifying one
level for each of the \( N \) treatments; \( X_1, \ldots, X_n \) are the \( n \) data points
as specified on the Equal Card.

FORM 2: \[ I_1, \ldots, I_N \ M \ X_1, \ldots, X_M \]
WHEN: cell frequencies are unequal
data for a cell on single card

\[ I_1, \ldots, I_N \] are integer cell-identification numbers, identifying one
level for each of the \( N \) treatments; \( M \) (an integer) is the number of
data points in that particular cell; \( X_1, \ldots, X_M \) are the \( M \) data points.

FORM 3: \[ I_1, \ldots, I_N \ J \ X_1, \ldots, X_m \]
WHEN: cell frequencies are equal
more than 1 card necessary for all data
points in each cell

\[ I_1, \ldots, I_N \] are integer cell-identification numbers, identifying one
level for each of the \( N \) treatments; \( J \) (integer) is the card sequence
number; \( X_1, \ldots, X_m \) are the \( m \) data points per sequenced card, as spec-
cified on the Equal Card.

FORM 4: \[ I_1, \ldots, I_N \ M \ J \ X_1, \ldots, X_m \]
WHEN: cell frequencies are unequal
more than 1 card necessary for all data
points in each cell

\[ I_1, \ldots, I_N \] are integer cell-identification numbers, identifying one
level for each of the \( N \) treatments; \( M \) (integer) is the number of
data points in that particular cell and must be repeated for every
card in that cell; \( J \) (integer) is the card sequence number; \( X_1, \ldots, X_m \) are the \( m \) data points per sequenced card, as specified on the Unequal Card.

Description of Output

For each cell, mean, number, variance, degrees of freedom, and cell
identification are printed at the option of the user. Chi square,
correction factor, degrees of freedom, and probability are printed
for the Bartlett's test for homogeneity of variance. For each treat-
ment group and combination, total, number of observations, mean, and
sum of squares are given at the option of the user. An analysis-of-
continued
variance summary table, including treatment names (according to Treatment-Names Card), sums of squares, degrees of freedom, mean squares, and F ratios, and their probabilities will be given, along
with the output from the Scheffe's test if applicable.

REFERENCES

SAMPLE INPUT

```
FREQUENCY DISTRIBUTION OF ANIMAL -1.0 SAMPLE OBSERVATIONS

TREATMENT SUMMARY

VARIABLES

CELL SUMMARY

TREATMENT MEANS: VAR Y, Z

SUMMARY(SI, F, G)

DATA FILE A

FILE 7

BEGIN DATA

1112143
1122143
1211112
1211112
71117
71117
221 4
222 A N
FIL DATA
END
```
**DEMONSTRATION OF ANOVA WITH EQUAL OBSERVATIONS PER CELL**

<table>
<thead>
<tr>
<th>VARIABLE NUMBER</th>
<th>N</th>
<th>MEAN</th>
<th>VARIANCE</th>
<th>DEGREES OF FREEDOM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>20.500</td>
<td>68.000</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>14.000</td>
<td>2.000</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>11.000</td>
<td>2.000</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>8.500</td>
<td>3.000</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>9.000</td>
<td>8.000</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>4.500</td>
<td>2.000</td>
<td>1</td>
</tr>
</tbody>
</table>

**DEGREES OF FREEDOM**

<table>
<thead>
<tr>
<th>CELL</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
</table>

**CORRECTION FACTOR IS**

1.3750

**BARTLETT'S TEST OF HOMOGENEITY OF VARIANCE:**

Chi Square = 9.5680

Prorability = 0.214402

*Use caution in interpretation if any cell frequencies are less than 3*
DEMONSTRATION OF ANOVA WITH EQUAL OBSERVATIONS PER CELL

Cell frequencies are equal - standard analysis follows

<table>
<thead>
<tr>
<th>VARIABLE NUMBER</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>TREATMENT LEVEL</td>
<td>TOTAL</td>
</tr>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>.11.0000</td>
</tr>
<tr>
<td>2</td>
<td>.56.0000</td>
</tr>
<tr>
<td>2</td>
<td>164.0000</td>
</tr>
<tr>
<td>2</td>
<td>74.0000</td>
</tr>
<tr>
<td>3</td>
<td>115.0000</td>
</tr>
<tr>
<td>2</td>
<td>126.0000</td>
</tr>
<tr>
<td>12</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>133.0000</td>
</tr>
<tr>
<td>2</td>
<td>34.0000</td>
</tr>
<tr>
<td>1</td>
<td>50.0000</td>
</tr>
<tr>
<td>2</td>
<td>23.0000</td>
</tr>
<tr>
<td>13</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>85.0000</td>
</tr>
<tr>
<td>2</td>
<td>26.0000</td>
</tr>
<tr>
<td>1</td>
<td>94.0000</td>
</tr>
<tr>
<td>2</td>
<td>32.0000</td>
</tr>
<tr>
<td>23</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>78.0000</td>
</tr>
<tr>
<td>2</td>
<td>37.0000</td>
</tr>
<tr>
<td>1</td>
<td>90.0000</td>
</tr>
<tr>
<td>2</td>
<td>36.0000</td>
</tr>
<tr>
<td>123</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>61.0000</td>
</tr>
<tr>
<td>2</td>
<td>17.0000</td>
</tr>
<tr>
<td>1</td>
<td>26.0000</td>
</tr>
<tr>
<td>2</td>
<td>9.0000</td>
</tr>
<tr>
<td>1</td>
<td>72.0000</td>
</tr>
<tr>
<td>2</td>
<td>19.0000</td>
</tr>
<tr>
<td>1</td>
<td>22.0000</td>
</tr>
<tr>
<td>2</td>
<td>14.0000</td>
</tr>
</tbody>
</table>

continued
**DEMONSTRATION OF ANOVA WITH EQUAL OBSERVATIONS PER CELL**

**VARIABLE NUMBER 1**

<table>
<thead>
<tr>
<th>Source</th>
<th>Sum of Squares</th>
<th>DF</th>
<th>Mean Squares</th>
<th>F Ratio</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEX</td>
<td>0.476563003</td>
<td>1</td>
<td>0.476563003</td>
<td>45.029</td>
<td>0.0000</td>
</tr>
<tr>
<td>INCOME</td>
<td>0.164646030</td>
<td>1</td>
<td>0.164646030</td>
<td>26.009</td>
<td>0.0000</td>
</tr>
<tr>
<td>JOB</td>
<td>0.256250001</td>
<td>1</td>
<td>0.256250001</td>
<td>26.009</td>
<td>0.0000</td>
</tr>
<tr>
<td>JD</td>
<td>0.315625000</td>
<td>1</td>
<td>0.315625000</td>
<td>31.527</td>
<td>0.0000</td>
</tr>
<tr>
<td>JD</td>
<td>0.315625000</td>
<td>1</td>
<td>0.315625000</td>
<td>31.527</td>
<td>0.0000</td>
</tr>
<tr>
<td>JD</td>
<td>0.315625000</td>
<td>1</td>
<td>0.315625000</td>
<td>31.527</td>
<td>0.0000</td>
</tr>
<tr>
<td>JD</td>
<td>0.315625000</td>
<td>1</td>
<td>0.315625000</td>
<td>31.527</td>
<td>0.0000</td>
</tr>
<tr>
<td>JD</td>
<td>0.315625000</td>
<td>1</td>
<td>0.315625000</td>
<td>31.527</td>
<td>0.0000</td>
</tr>
<tr>
<td>JD</td>
<td>0.315625000</td>
<td>1</td>
<td>0.315625000</td>
<td>31.527</td>
<td>0.0000</td>
</tr>
<tr>
<td>ERROR</td>
<td>0.173500003</td>
<td>8</td>
<td>0.021687502</td>
<td>1.271</td>
<td>0.2923</td>
</tr>
</tbody>
</table>

Corrected Total Sum of Squares = 0.207494004

Uncorrected Total Sum of Squares = 0.570500004

<table>
<thead>
<tr>
<th>Treatment</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Difference</th>
<th>Scheffe's Constant</th>
<th>Estimate of Variance of Differences</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0.1563002</td>
<td>2.316</td>
<td>0.23280 01</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0.1188002</td>
<td>2.316</td>
<td>0.23280 01</td>
</tr>
</tbody>
</table>
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 8 seconds on the central-processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
               = $0.88 + network overhead

CONTENTS—(STPAC) ANOVUM

<table>
<thead>
<tr>
<th>pages</th>
<th>Identification &amp; Abstract</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3-7</td>
<td>User Instructions</td>
</tr>
<tr>
<td>9-12</td>
<td>I/O</td>
</tr>
<tr>
<td>13</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>
ANOVES is designed to handle a factorial analysis of variance with unequal numbers of observations in subclasses (cells). A problem with equal subclass numbers is acceptable, although it is not advisable to use ANOVES for a balanced design with equal subclass numbers owing to its relatively long computing time as compared to that of ANOVUM. Up to 150 variables (separate analyses) will be permitted with data code = 1. Only one variable is acceptable with data code = 0.

This method of analyzing data of multiple classifications with unequal subclass numbers is based on the assumption that the population from which the sample is drawn has proportional or equal subclass numbers. Under its fundamental hypotheses, this method affords an estimate of both the main effects and the interactions. However, every subclass must contain at least one observed value.

If the computed chi-square value for disproportionality is significant, a reduction factor will be calculated and all the sums of squares (except the with-in subclasses, which is calculated from the original data) will be reduced accordingly. This is an attempt to remove part of the disproportionality of the data.
The probability associated with the chi-square value computed by using subclass numbers is calculated and the 0.05 probability point is considered critical in determining whether a reduction factor will be computed or not.

The reduction factor is

\[
\frac{N}{\sum_{i=1}^{K} \frac{E_i^2}{A_i}}
\]

where \(N\) is the total number of observations, \(K\) is the number of cell subclasses, \(E\) is the expected number of observations in a cell, and \(A\) is the actual number of observations in the same cell.

A Bartlett's test for homogeneity of variance is performed on the raw data. If the chi square computed in this test is significant, the conclusions of the analysis of variance should be interpreted cautiously.

REFERENCES


Bennett, K.R. (personal correspondence to J. Streeter).

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute and Format Cards are essential for ANOVES. Begin- and End-Data Cards are also necessary if the data have not already been read for a previous STPAC program. All other STPAC Control Cards are optional. Some additional Control Cards are required and some optional Control Cards are specific to ANOVES (see below).

Treatments Card (required)

Columns | Contents
---|---
1-10 | TREATMENTS
12 | \( N \), the number of treatments
14-19 | LEVELS
21-80 | \( L_1, L_2, \ldots, L_N \); \( L_i \) is the number of levels for the \( i \)th treatment

Data-Code Card (required)

Columns | Contents
---|---
1-9 | DATA CODE
11 | 1: there is one data point per variable per card, preceded by cell identification
0: all data for a single cell are punched on a single card or card sequence. If more than one Data Card per cell is required (i.e., if cell identification and data use more than 80 columns), a sequence number must be included on each card, following the cell identification. See Data Cards, below.

Equal or Unequal Card (required only if data code is 0)

Columns | Contents
---|---
1-80 | EQUAL \( n \) SEQUENCED \( m \) PERCARD
or

continued
Columns Contents
UNEQUAL n SEQUENCED m PERCARD

This card indicates whether the cell frequencies are equal or unequal. n must be the number of observations per cell if cells are equal or the maximum number of observations per cell if cells are unequal.

The next three fields may be omitted if only one Data Card per cell is necessary. If, however, more than one Data Card per cell is necessary, the last three fields of this card must be included. m, in this case, is the number of data points per card.

Punch Card (optional)
Columns Contents
1-5 PUNCH

This card should be included only if the user desires punched-card output of means.

Cell Summary Card (optional)
Columns Contents
1-12 CELL SUMMARY

This card calls for a summary table of cell frequencies, means, variance, and degrees of freedom to be printed.

Treatment Summary Card (optional)
Columns Contents
1-17 TREATMENT SUMMARY

This card calls for a summary table of totals, number of observations, means, and sums of squares for each treatment group and combination to be printed.

Missing-Data Card (optional)
Columns Contents
1-12 MISSING DATA
14-21 x, any number up to eight digits, including a decimal, which will be read as a code for missing data

Include this card only if some data are missing. If this card is omitted, it will be assumed that all fields contain data.
will be read as zero). If this card is included withCols. 14–21 blank, zeros and blanks in the Data Cards will be treated as missing data. IfCols. 14–21 are punched, the number x in a data field will be treated as a case of missing data.

No standard format is provided for this program; format must be supplied by the user as specified in the requirements for STPAC.

Data Cards

Data Cards are punched with code numbers signifying level for each treatment, starting at the left-hand side of the card (cell identification). These must be in integer format. The first (I₁) indicates level for treatment 1; the second (I₂) indicates level for treatment 2; etc. If any of the I₁, I₂, ..., Iₙ are blank or zero, the card will be ignored by the program. If any of the I₁, I₂, ..., Iₙ are greater than the maximum number of levels specified on the Treatments Card, the analysis will not be performed.

If the data code is 1, a single data point for each of the variables specified (in F, D, or E format) follows the cell identification. Data Cards with this data code may be in any order.

If the data code is 0, and cell frequencies are unequal, the number of data points in that particular cell (in I format) must immediately follow the cell identification. If a sequence number is also required, it (also in I format) follows the number of data points. In the case where more than one card per cell is needed under data code = 0, cell identification, number of data points, and sequence number must be the first items of information on every card. Data fields follow the preceding fields. All cards for a single cell must be together.

Data Code = 1 FORM: \[ I₁, \ldots, Iₙ X₁, \ldots, Xₖ \]
where I₁, ..., Iₙ are integer cell-identification numbers, identifying one level for each of the N treatments; X₁, ..., Xₖ are data points, one for each of K specified variables.

Data Code = 0 FORM 1: \[ I₁, \ldots, Iₙ X₁, \ldots, Xₙ \]
WHEN: cell frequencies are equal all data for a cell on single card

I₁, ..., Iₙ are integer cell-identification numbers, identifying one level for each of the N treatments; X₁, ..., Xₙ are the n data points as specified on the Equal Card.
FORM 2: \( I_1, I_2, \ldots, I_N \quad M \quad X_1, X_2, \ldots, X_M \)

WHEN: cell frequencies are unequal
data for a cell on single card

\( I_1, I_2, \ldots, I_N \) are integer cell-identification numbers, identifying one level for each of the \( N \) treatments; \( M \) (an integer) is the number of data points in that particular cell; \( X_1, X_2, \ldots, X_M \) are the \( M \) data points.

FORM 3: \( I_1, I_2, \ldots, I_N \quad J \quad X_1, X_2, \ldots, X_m \)

WHEN: cell frequencies are equal
more than 1 card necessary for all data points in each cell

\( I_1, I_2, \ldots, I_N \) are integer cell-identification numbers, identifying one level for each of the \( N \) treatments; \( J \) (integer) is the card sequence number; \( X_1, X_2, \ldots, X_m \) are the \( m \) data points per sequenced card, as specified on the Equal Card.

FORM 4: \( I_1, I_2, \ldots, I_N \quad M \quad J \quad X_1, X_2, \ldots, X_m \)

WHEN: cell frequencies are unequal
more than 1 card necessary for all data points in each cell

\( I_1, I_2, \ldots, I_N \) are integer cell-identification numbers, identifying one level for each of the \( N \) treatments; \( M \) (integer) is the number of data points in that particular cell and must be repeated for every card in that cell; \( J \) (integer) is the card sequence number; \( X_1, X_2, \ldots, X_m \) are the \( m \) data points per sequenced card, as specified on the Unequal Card.

Description of Output

The chi-square value of the Bartlett's test for homogeneity of variance is printed out, along with its associated probability. For each cell, the total, number of observations, mean, expected total, and expected mean are printed also. The chi-square value for disproportionality is printed along with its associated probability. If this chi square is significant at the 0.05 level, the reduction factor is also printed. For each treatment group and combination, expected total, expected number of observations, and sum of squares are given. An analysis-of-variance summary table, including treatment names (according to Treatment-Names Card), sums of squares, degrees of freedom, mean squares, and F ratios, is printed last. Card output of means is provided at the option of the user.
SAMPLE INPUT

EXFC ANNIVS
TREATMENTS 1-9
DATA CODE 1
TREATMENT MANPS TCNFERA
FORMAT 41,F2.7
TREATMENT SUMMARY
VARIABLES 1
AR4I DATA
11129
11229
12314
12310
211 0
2711
221 4
222 A
11321
11292
22311
22214
END DATA
STOP
BARTLETT'S TEST FOR HOMOGENEITY OF VARIANCE

BARTLETT'S TEST CANNOT BE COMPUTED BECAUSE ONE OR MORE CELL FREQUENCIES EQUAL 1;
OR BECAUSE ONE OR MORE VARIANCES EQUAL 0.

CHI SQUARE OF DISPROPORTIONALITY OF SUBCLASS FREQUENCIES = 1.3333 P = 0.248213

TREATMENT SUMMARY - MARGINAL VALUES

<table>
<thead>
<tr>
<th>FACTOR</th>
<th>LEVEL</th>
<th>EXPECTED TOTAL</th>
<th>EXPECTED NO.</th>
<th>EXPECTED MEAN</th>
<th>SUM OF SQUARES</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>129.7500</td>
<td>6.000</td>
<td>20.1250</td>
<td>0.2430090 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>59.2500</td>
<td>6.000</td>
<td>9.8750</td>
<td>0.5859440 03</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>113.7500</td>
<td>6.000</td>
<td>18.8750</td>
<td>0.2137590 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>66.7500</td>
<td>6.000</td>
<td>11.1250</td>
<td>0.7429440 03</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>93.0000</td>
<td>6.200</td>
<td>15.5000</td>
<td>0.1441500 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>87.0000</td>
<td>6.000</td>
<td>14.5000</td>
<td>0.1261500 04</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>83.2500</td>
<td>3.000</td>
<td>27.7500</td>
<td>0.2310190 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>30.3000</td>
<td>3.000</td>
<td>10.0000</td>
<td>0.3000000 03</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>37.5000</td>
<td>3.000</td>
<td>12.5000</td>
<td>0.6687500 03</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>29.2500</td>
<td>3.000</td>
<td>9.7500</td>
<td>0.2651880 03</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>64.5000</td>
<td>3.000</td>
<td>21.5000</td>
<td>0.1386750 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>28.5000</td>
<td>3.200</td>
<td>9.5000</td>
<td>0.2707500 03</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>96.2500</td>
<td>3.200</td>
<td>18.7500</td>
<td>0.1054490 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>90.7500</td>
<td>3.000</td>
<td>10.2500</td>
<td>0.3151880 03</td>
</tr>
<tr>
<td>23</td>
<td>1</td>
<td>55.5000</td>
<td>3.000</td>
<td>18.5000</td>
<td>0.1026750 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>37.5000</td>
<td>3.200</td>
<td>12.5000</td>
<td>0.6687500 03</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>97.7500</td>
<td>3.200</td>
<td>19.2500</td>
<td>0.1311690 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>92.2500</td>
<td>3.000</td>
<td>9.7500</td>
<td>0.2651880 03</td>
</tr>
<tr>
<td>123</td>
<td>1</td>
<td>42.0000</td>
<td>1.500</td>
<td>28.0000</td>
<td>0.1176000 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>13.5000</td>
<td>1.500</td>
<td>9.0000</td>
<td>0.1215000 03</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>22.5000</td>
<td>1.500</td>
<td>15.0000</td>
<td>0.3375000 03</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>15.0000</td>
<td>1.500</td>
<td>10.0000</td>
<td>0.1500000 03</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>41.2500</td>
<td>1.500</td>
<td>27.0000</td>
<td>0.1136380 04</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>16.5000</td>
<td>1.500</td>
<td>11.0000</td>
<td>0.1815000 03</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>15.0000</td>
<td>1.500</td>
<td>10.0000</td>
<td>0.1500000 03</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>14.2500</td>
<td>1.500</td>
<td>9.5000</td>
<td>0.1353750 03</td>
</tr>
</tbody>
</table>

ANALYSIS OF VARIANCE SUMMARY TABLE

<table>
<thead>
<tr>
<th>SOURCE</th>
<th>SUMS OF SQUARES</th>
<th>DF</th>
<th>MEAN SQUARES</th>
<th>F RATIO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.3151880 03</td>
<td>1</td>
<td>0.3151880 03</td>
<td>12.997</td>
</tr>
<tr>
<td>2</td>
<td>0.1801880 03</td>
<td>1</td>
<td>0.1801880 03</td>
<td>7.430</td>
</tr>
<tr>
<td>3</td>
<td>0.1687500 03</td>
<td>1</td>
<td>0.1687500 03</td>
<td>0.124</td>
</tr>
<tr>
<td>12</td>
<td>0.1687500 03</td>
<td>1</td>
<td>0.1687500 03</td>
<td>6.959</td>
</tr>
<tr>
<td>13</td>
<td>0.9187500 01</td>
<td>1</td>
<td>0.9187500 01</td>
<td>0.379</td>
</tr>
<tr>
<td>23</td>
<td>0.9187500 01</td>
<td>1</td>
<td>0.9187500 01</td>
<td>0.379</td>
</tr>
<tr>
<td>123</td>
<td>0.7500000 00</td>
<td>1</td>
<td>0.7500000 00</td>
<td>0.031</td>
</tr>
<tr>
<td>ERROR</td>
<td>0.9700000 00</td>
<td>4</td>
<td>0.2425000 02</td>
<td></td>
</tr>
</tbody>
</table>

'INCORRECTED TOTAL SUM OF SQUARES = 0.4085000 04
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 4 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
= $0.44 + network overhead
AOVRM is designed to handle a factorial analysis of variance with repeated measures and either equal or unequal numbers of observations in subclasses (cells). A problem with no repeated measures is acceptable, although it is not advisable to use AVORM for such a design owing to the relatively long computing time as compared with that of ANOVM or ANOVES.

This method of analyzing data of multiple classifications with unequal subclass numbers is based on the assumption that the population from which the sample is drawn has proportional or equal subclass numbers. This method allows calculation of all main effects and interactions. However, every subclass must contain at least one subject and the data for every subject included must be complete.

REFERENCES

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute, Variables, and Format Cards are essential for AVORM. Begin- and End-Data Cards are also necessary if the data have not already been read for a previous STPAC program. All other STPAC Control Cards are optional. Some additional Control Cards are required and some optional Control Cards are specific to AOVRM (see below).

Treatments Card (required)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-10</td>
<td>TREATMENTS</td>
</tr>
<tr>
<td>12</td>
<td>N, the number of treatments</td>
</tr>
<tr>
<td>14-19</td>
<td>LEVELS</td>
</tr>
<tr>
<td>21-80</td>
<td>L₁, L₂, ⋯, Lₙ; Lᵢ is the number of levels for the iᵗʰ treatment</td>
</tr>
</tbody>
</table>

This card form is used only if no repeated measures are involved in the design. For repeated treatments, use the following form.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-10</td>
<td>TREATMENTS</td>
</tr>
<tr>
<td>12</td>
<td>N, the total number of treatments</td>
</tr>
<tr>
<td>14-21</td>
<td>REPEATED</td>
</tr>
<tr>
<td>23</td>
<td>M, the number of repeated treatments</td>
</tr>
<tr>
<td>25-30</td>
<td>LEVELS</td>
</tr>
<tr>
<td>32-80</td>
<td>L₁, L₂, ⋯, Lₙ, as above. The repeated-measure treatments must be the last M of the N treatments</td>
</tr>
</tbody>
</table>

Data-Code Card (required)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-9</td>
<td>DATA CODE</td>
</tr>
</tbody>
</table>

continued
Data-Code Card

Columns  Contents
11    1: there is one data point per card, preceded by cell identification for all treatments, and the subject number
0: all data for a single subject are punched on a single card or card sequence. If more than one Data Card per subject is required (i.e., if cell identification and data use more than 80 columns), the format supplied by the user must take this into account. The program will not. Cell identification in this case is only for the nonrepeated measures and the subject number

Observations Card (required)

Columns  Contents
1-12    OBSERVATIONS
14-80    N, an integer, the number of observations per cell (in the case of equal cell frequencies) or the maximum number of subjects per cell (in the case of unequal cell frequencies)

Punch Card (optional)

Columns  Contents
1-5    PUNCH
This card should be included only if the user desires punched-card output of means.

Cell Summary Card (optional)

Columns  Contents
1-12    CELL SUMMARY
This card calls for a summary table of cell frequencies, means, variance, and degrees of freedom to be printed.

Treatment Summary Card (optional)

Columns  Contents
1-17    TREATMENT SUMMARY
This card calls for a summary table of totals, number of observations, means, and sums of squares for each treatment group and combination to be printed.
Missing-Data Card (required if there are unequal number of subjects in subclasses)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-12</td>
<td>MISSING DATA</td>
</tr>
<tr>
<td>14-21</td>
<td>x, any number up to eight digits, including a decimal, which will be read as a code for missing data</td>
</tr>
</tbody>
</table>

If this card is omitted, it will be assumed that all fields contain data (blanks will be read as zero). If this card is included with Cols. 14-21 blank, zeros and blanks in the Data Cards will be treated as missing data. If Cols. 14-21 are punched, the number x in a data field will be treated as a case of missing data.

Data Cards

If data code = 1, Data Cards are punched with code numbers signifying level for each treatment, the code for the subject (all in integer format), followed by a single data point (in F, E, or D format).

If data code = 0, Data Cards are punched with code numbers signifying level for each nonrepeated treatment, the code for the subject (all in integer format), followed by all the data for that subject (in F, E, or D format).

The order in which the data are punched if data code = 0 is critical, since order is the only way in which the program can identify the data. The first treatment level to be changed should be the last repeated-measure treatment. The last to be changed should be the first repeated-measure treatment; i.e., data for the levels of the innermost treatment in the nest of treatments should be punched first, and so on, to the outermost.

Subject codes must be consecutive, beginning with 1, for each individual cell, whether data code = 0 or 1.

Description of Output

For each cell, the mean, number, variance, degrees of freedom, and cell identification are printed at the option of the user. For each treatment group and combination, the total, number of observations, mean, and sum of squares are given, also at the user's option.

An analysis-of-variance summary table—Including treatment names (according to the Treatment-Names Card, if provided), sums of squares, degrees of freedom, mean squares, and F ratios for all treatments, interactions, and error terms—is printed last.

Punched-card output of cell means is provided optionally.
SAMPLE INPUT

1. TREATMENTS & APPLICATIONS
2. DATA CUMULUS
3. TREATMENT SUMMARY

VARIABLES:
1. ID
2. TREATMENT NAMES
3. REGISTRATION DATA

BEGIN DATA
1184.329.06.00.0
1234.349.01.06.0
1234.349.01.06.00
2143.434.07.06.0
2143.434.07.06.0
2143.434.07.06.0
END DATA
### SAMPLE OUTPUT

<table>
<thead>
<tr>
<th>TREATMENT</th>
<th>LEVEL</th>
<th>TOTAL</th>
<th>N</th>
<th>MEAN</th>
<th>SUM OF SQUARES</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1375.0000</td>
<td>24</td>
<td>44.7917</td>
<td>0.4915100 05</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1311.0000</td>
<td>24</td>
<td>42.1250</td>
<td>0.4258840 05</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1024.0000</td>
<td>24</td>
<td>42.6667</td>
<td>0.4169070 05</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1060.0000</td>
<td>24</td>
<td>44.2500</td>
<td>0.4699350 05</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>720.0000</td>
<td>16</td>
<td>45.0000</td>
<td>0.3240000 05</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>706.0000</td>
<td>16</td>
<td>44.1250</td>
<td>0.3115230 05</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>660.0000</td>
<td>16</td>
<td>41.2500</td>
<td>0.2722500 05</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>912.0000</td>
<td>24</td>
<td>38.9000</td>
<td>0.3465600 05</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1174.0000</td>
<td>24</td>
<td>48.9167</td>
<td>0.5742820 05</td>
</tr>
</tbody>
</table>

...continued
### ANALYSIS OF VARIANCE SUMMARY TABLE

#### BETWEEN SUBJECTS

<table>
<thead>
<tr>
<th>Source</th>
<th>Sums of Squares</th>
<th>DF</th>
<th>Mean Squares</th>
<th>F Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEX</td>
<td>0.8533330</td>
<td>2</td>
<td>0.4266667</td>
<td>11.5000</td>
</tr>
<tr>
<td>OCCUPATI</td>
<td>0.3008330</td>
<td>1</td>
<td>0.3008330</td>
<td>4.0700</td>
</tr>
<tr>
<td>ERROR</td>
<td>0.2950000</td>
<td>4</td>
<td>0.0737500</td>
<td>1.1300</td>
</tr>
</tbody>
</table>

#### WITHIN SUBJECTS

<table>
<thead>
<tr>
<th>Source</th>
<th>Sums of Squares</th>
<th>DF</th>
<th>Mean Squares</th>
<th>F Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERIOD</td>
<td>0.1251670</td>
<td>3</td>
<td>0.0416687</td>
<td>29.8000</td>
</tr>
<tr>
<td>13</td>
<td>0.3166670</td>
<td>01</td>
<td>0.3166670</td>
<td>3.9000</td>
</tr>
<tr>
<td>24</td>
<td>0.2066670</td>
<td>02</td>
<td>0.1033330</td>
<td>5.0100</td>
</tr>
<tr>
<td>ERROR</td>
<td>0.1650000</td>
<td>02</td>
<td>0.0825000</td>
<td>0.7600</td>
</tr>
<tr>
<td>TEST</td>
<td>0.1430080</td>
<td>4</td>
<td>0.3580000</td>
<td>162.664</td>
</tr>
<tr>
<td>14</td>
<td>0.1633333</td>
<td>02</td>
<td>0.0816667</td>
<td>1.9500</td>
</tr>
<tr>
<td>24</td>
<td>0.3533330</td>
<td>00</td>
<td>0.3533330</td>
<td>0.0300</td>
</tr>
<tr>
<td>ERROR</td>
<td>0.3516670</td>
<td>02</td>
<td>0.1758333</td>
<td>0.0960</td>
</tr>
<tr>
<td>34</td>
<td>0.3816670</td>
<td>02</td>
<td>0.1908333</td>
<td>5.4850</td>
</tr>
<tr>
<td>134</td>
<td>0.6666670</td>
<td>00</td>
<td>0.3333330</td>
<td>0.0960</td>
</tr>
<tr>
<td>234</td>
<td>0.1850000</td>
<td>00</td>
<td>0.0925000</td>
<td>2.6500</td>
</tr>
<tr>
<td>1234</td>
<td>0.3166670</td>
<td>01</td>
<td>0.1583333</td>
<td>0.4500</td>
</tr>
<tr>
<td>ERROR</td>
<td>0.2783330</td>
<td>32</td>
<td>0.0470170</td>
<td>0.1500</td>
</tr>
</tbody>
</table>

**GRAND TOTAL = 2.3086000**

**NUMBER OF DATA POINTS = 48**

**CORRECTION FACTOR = 0.9065410**

**UNCORRECTED SUM OF SQUARES = 0.4255000**

---

**continued**

3/70
<table>
<thead>
<tr>
<th>ACCOUNT NUMBER</th>
<th>DATE PROCESSED</th>
<th>NAME OF PERSON REQUESTING RUN</th>
<th>USER IDENT.</th>
<th>MAXIMUM TIME</th>
<th>ACTUAL TIME</th>
<th>MAXIMUM RECORDS</th>
<th>ACTUAL RECORDS</th>
<th>COST</th>
<th>JOB NAME</th>
<th>SITE CAT. SERIAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>$4002</td>
<td>12/08/69</td>
<td>DAUBERT N C</td>
<td>ADVRM</td>
<td>100</td>
<td>9</td>
<td>2000</td>
<td>342</td>
<td>$0.00</td>
<td>AA</td>
<td>3</td>
</tr>
</tbody>
</table>
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 5 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
= $0.55 + network overhead

CONTENTS—(STPAC) AOVRM

pages
1 Identification & Abstract
3-5 User Instructions
7-10 I/O
11 Cost—Contents
BARTL performs a Bartlett's test for homogeneity of variance on up to 6000 subgroups (or cells). Subgroups may be equal or unequal.

REFERENCES

USER INSTRUCTIONS

This program is one of a collection of programs that are all executed using the library program STPAC. The writeup for STPAC must be consulted for the description of additional Control Cards and procedures for running this program.

Control Cards and Their Functions

Of the various Control Cards discussed in the STPAC writeup, the Execute and Format Cards are essential for BARTL. Begin- and End-Data Cards are also necessary if the data have not already been read for a previous STPAC program. All other STPAC Control Cards are optional. Some additional Control Cards are required and some optional Control Cards are specific to BARTL (see below).

Treatments Card (required only if data code = 0 or 1—i.e., data are punched as for analysis of variance)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-10</td>
<td>TREATMENTS</td>
</tr>
<tr>
<td>12</td>
<td>N, the number of treatments</td>
</tr>
<tr>
<td>14-19</td>
<td>LEVELS</td>
</tr>
<tr>
<td>21-80</td>
<td>L_1, L_2, ..., L_N; L_i is the number of levels for the i^{th} treatment</td>
</tr>
</tbody>
</table>

Data-Code Card (required)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-9</td>
<td>DATA CODE</td>
</tr>
<tr>
<td>11</td>
<td>1 or 2: there is one data point per card, preceded by subgroup identification. Further differentiation between these two codes will be included in the section on Data Cards. 0: all data for a single subgroup are punched on a single card or card sequence. If more than one Data Card per subgroup is required (i.e., if subgroup identification and data use more than 80 columns), a sequence number must be included on each card, following the subgroup identification. See Data Cards, below.</td>
</tr>
</tbody>
</table>

continued
Equal or Unequal Card (required only if data code is 0)

**Columns**

**Contents**

1–80  
EQUAL n SEQUENCED m PERCARD  

or  
UNEQUAL n SEQUENCED m PERCARD

This card indicates whether the cell frequencies are equal or unequal. \( n \) must be the number of observations per cell if cells are equal or the maximum number of observations per cell if cells are unequal.

The next three fields may be omitted if only one Data Card per cell is necessary. If, however, more than one Data Card per cell is necessary, the last three fields of this card must be included. \( m \), in this case, is the number of data points per card.

No standard format is provided for this program; format must be supplied by the user as specified in the requirements for STPAC.

**Data Cards**

Data Cards are punched with code numbers that are the subgroup- or cell-identification numbers. These must be in integer format.

For data code = 2, the subgroup identification must be a single-integer number, and a single data point (in F, E, or D format) follows the identification number. Identification numbers in this data code need not be sequential (i.e., an analysis could be performed on subgroups identified as 1, 22, 1000, 50), but all cards for a single subgroup must be together.

For data codes = 0 and 1, Data Cards are punched with code numbers signifying level for each treatment, starting at the lefthand side of the card (cell identification). These must be in integer format. The first (\( I_1 \)) indicates level for treatment 1; the second (\( I_2 \)) indicates level for treatment 2; etc.

If the data code is 1, a single data point (in F, D, or E format) follows the cell identification (cell identification as in the analysis of variance). Data Cards with this data code may be in any order.

**Data Code = 2**

FORM:  
\[ I \times \]

where \( I \) is an integer subgroup-identification number and \( X \) is a single data point in F, E, or D format. This is the proper form whether cell frequencies are equal or unequal.

continued
Data Code = 1

FORM: \[ I_1, \ldots, I_N, X \]

where \( I_1, \ldots, I_N \) are integer cell-identification numbers, identifying one level for each of the \( N \) treatments; \( X \) is a single data point.

Data Code = 0

FORM 1: \[ I_1, \ldots, I_N, X_1, \ldots, X_n \]

WHEN: cell frequencies are equal

all data for a cell on single card

\( I_1, \ldots, I_N \) are integer cell-identification numbers, identifying one level for each of the \( N \) treatments; \( X_1, \ldots, X_n \) are the \( n \) data points as specified on the Equal Card.

FORM 2: \[ I_1, \ldots, I_N, M, X_1, \ldots, X_m \]

WHEN: cell frequencies are unequal

data for a cell on single card

\( I_1, \ldots, I_N \) are integer cell-identification numbers, identifying one level for each of the \( N \) treatments; \( M \) (an integer) is the number of data points in that particular cell; \( X_1, \ldots, X_m \) are the \( M \) data points.

FORM 3: \[ I_1, \ldots, I_N, J, X_1, \ldots, X_m \]

WHEN: cell frequencies are equal

more than 1 card necessary for all data points in each cell

\( I_1, \ldots, I_N \) are integer cell-identification numbers, identifying one level for each of the \( N \) treatments; \( J \) (integer) is the card sequence number; \( X_1, \ldots, X_m \) are the \( m \) data points per sequenced card, as specified on the Equal Card.

FORM 4: \[ I_1, \ldots, I_N, M, J, X_1, \ldots, X_m \]

WHEN: cell frequencies are unequal

more than 1 card necessary for all data points in each cell

\( I_1, \ldots, I_N \) are integer cell-identification numbers, identifying one level for each of the \( N \) treatments; \( M \) (integer) is the number of data points in that particular cell and must be repeated for every card in that cell; \( J \) (integer) is the card sequence number; \( X_1, \ldots, X_m \) are the \( m \) data points per sequenced card, as specified on the Unequal Card.

continued
Description of Output
For each subgroup (or cell), its variance, degrees of freedom, and identification are printed.

The chi-square value of the Bartlett's test, along with its correction factor, degrees of freedom, and associated probability are printed.

NOTE: A Bartlett's test cannot and will not be calculated if any of the subgroups have only one observation or a variance equal to 0.
SAMPLE INPUT

```
PYCINIT RANMI
CRLI VLSRW
TITLE:  HOLLAND F. W. NICE
PRICE:  11.67, 24.43
DATA FROM I
TREATMENTS:  1, 2, 3
BEGIN DATA
1 1 2.0
1 1 2.0
1 1 4.3
1 1 2.7
1 1 4.8
1 1 4.8
1 1 4.4
1 1 2.7
1 1 4.8
1 1 2.7
2 1 4.8
2 1 3.7
2 1 4.8
2 1 2.7
2 1 2.7
2 1 2.7
2 1 4.8
2 1 4.8
2 1 4.8
3 1 4.8
3 1 4.8

...`
BARTLETT'S TEST FOR HOMOGENEITY OF VARIANCE

<table>
<thead>
<tr>
<th>TREATMENT</th>
<th>POLAND</th>
<th>CHINA</th>
<th>PIGS</th>
<th>DEGREES OF FREEDOM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.998</td>
<td>0.989</td>
<td>0.988</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.4569</td>
<td>0.551</td>
<td>0.500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0889</td>
<td>0.1021</td>
<td>0.1466</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.3536</td>
<td>0.3500</td>
<td>0.3000</td>
<td></td>
</tr>
</tbody>
</table>

CORRECTION FACTOR IS 1.07395

BARTLETT'S TEST OF HOMOGENEITY OF VARIANCE:

CHI SQUARE = 17.8350

PR (PROBABILITY) = 0.012737

7 DEGREES OF FREEDOM

CAUTION: IN DEGREES OF FREEDOM IF ANY CELL FREQUENCIES ARE LESS THAN 5

CHI SQUARE IS SIGNIFICANT AT THE .05 LEVEL.
DATA DO NOT SUPPORT THE HYPOTHESIS OF HOMOGENEITY OF VARIANCE.
COST ESTIMATE

For the job listed on the Sample Input, the total running time was 4 seconds on the central processor unit, at $0.11 per second.

Charge to user = computer time + network overhead
               = $0.44 + network overhead

CONTENTS—(STPAC) BARTL

<table>
<thead>
<tr>
<th>pages</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Identification &amp; Abstract</td>
</tr>
<tr>
<td>3-6</td>
<td>User Instructions</td>
</tr>
<tr>
<td>7-8</td>
<td>I/O</td>
</tr>
<tr>
<td>9</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>
FUNCTIONAL ABSTRACT

The CalComp plot routines are a series of subroutines used to obtain graphic output. They are designed for the following purposes:

1) providing values to scale data to fit the plotting area,
2) drawing identification symbols at plotted data points,
3) drawing connected lines between data points (when desired),
4) drawing and labeling axes at any desired orientation, and
5) selecting from a large number of characters, numbers, and special symbols, any of which can be drawn in any size and at any angular orientation anywhere on the plotting surface.

These routines have been written by CalComp and modified by the University of Iowa to run under OS on the IBM 360/65. Plotting is done off line. The user's plotter program will create a plot tape containing various commands used to drive a drum plotter (see fig. 1). Each of eight possible commands moves the pen 1/100 of an inch in one of the eight designated directions (see fig. 2). In addition there are commands to move continued
The pen up and down. The plot tape is then taken to a smaller computer which drives the CalComp plotter (see fig. 3).

Access to the plot routines is by the FORTRAN "CALL" statement. Integer and floating point conventions must be followed (i.e. integer arguments must be used where the dummy argument specifies integer, likewise for floating point). All x and y coordinates used as arguments, must be expressed as floating point inches within actual page dimensions and in deflection from the (0.,0.) origin or the established reference point which is an established (0.,0.) origin. If the y coordinates are not within actual page dimensions in deflection from the reference point, that reference point will be destroyed and anything plotted after that will be erroneous. For orientation of the paper see fig. 1. All angles must be expressed in floating point degrees, with the positive sense counterclockwise. Character heights are specified as floating point inches. These heights should be greater than 0.07 but less than page size. Page size is 11 inches high and any reasonable length, since paper is continuous to a maximum length of 120 ft.

Figure 1
REFERENCES

Plotter Subroutines for CalComp Digital Incremental Plotter
(Iowa City, Ia.: Univ. Comp. Ctr., Univ. of Ia., 1970), Revised. Available from the EIN Office at the cost of reproduction and mailing.
USER INSTRUCTIONS

The user is referred to the reference below for instructions regarding use of the subroutines.

REFERENCES

SAMPLE INPUT

// EXEC FORTPLOT
//FORT.SYSIN DD *
   DIMENSION BUFFER(600),IDENT(9)
   DATA IDENT/' LOUISE LEVINE UCC PLOT COURSE '/
   C SET REFERENCE POINT PENUP
   CALL PLOT(0.,-12.,-3)
   C MOVE ORIGIN PEN UP
   CALL PLOT(0.,2.,-3)
   C DRAW BOX 8X3.5 PEN DOWN
   CALL PLOT(8.,0.,2)
   CALL PLOT(8.,3.5,2)
   CALL PLOT(0.,3.5,2)
   CALL PLOT(0.,0.0,2)
   C PRINT NAME IN BOX
   CALL SYMBOL(1.,2.,1.,'LOUISE',0.,6)
   CALL SYMBOL(1.,.5,1.,'LEVINE',0.,6)
   C TERMINATE PLOTTING
   CALL WRAPUP
   CALL EXIT
END
//GO.SYSIN DD *
/*
SAMPLE OUTPUT

REDUCED TO 60% OF ORIGINAL SIZE.

LOUISE LEVINE  UCC PLOT COURSE
COST ESTIMATE

The costing algorithm is based on the amount of computing time, I/O accesses, core storage, cards in and out, and lines and pages printed. The total cost for the Sample Output above was $0.71.

Charge to user = computer costs + postage + network overhead
= $0.71 + postage + network overhead

CONTENTS—CALCOMP PLOTTER SUBROUTINES

pages
1–3 Identification & Abstract
5 User Instructions
7–8 I/O
9 Cost—Contents

8/71
DESCRIPTIVE TITLE
BEEF Data Processing Subroutines

CALLING NAME
BEEFDP

INSTALLATION NAME
The University of Notre Dame Computing Center

AUTHOR(S) AND AFFILIATION(S)
Westinghouse Electric Corporation
UNIVAC Division of Sperry Rand Corporation

LANGUAGE
SLEUTH II/FORTRAN IV

COMPUTER
UNIVAC 1107

PROGRAM AVAILABILITY
Decks and listings presently available

CONTACT
Elizabeth Hutcheson, EIN Technical Representative, Computing Center, University of Notre Dame, Notre Dame, Ind. 46556
Tel.: (219) 283-7784

FUNCTIONAL ABSTRACT
The BEEF Data Processing library is a set of subroutines supplied by UNIVAC to "enhance FORTRAN's abilities as a commercial processor." This enrichment is in the form of subroutines for whole word data movement, character and field movement, supplementary formatting, decision-making with FORTRAN arrays, data conversion, report generation and control, I/O control, sorting, and compatibility with EAM/EDP (Electronic Accounting Machine/Electronic Data Processing) equipment.

REFERENCE
UNIVAC 1107 BEEF Data Processing Manual, (UP-3985), (UNIVAC Division of Sperry Rand Corp., N.Y.). Copies of this manual are available through the local UNIVAC representative or through the EIN Office at the cost of reproduction and mailing.
USER INSTRUCTIONS

Full details of the description and use of the BEEFDP library are given in the reference cited below. A brief description of the library subroutines is presented as an aid for the interested person. The user should specify the subprogram name to obtain all the subroutines listed under that name, for a run.

<table>
<thead>
<tr>
<th>Subprogram Name</th>
<th>Subroutine Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WWDM</td>
<td>MOVEKA</td>
<td>Moves the contents of a word into an array</td>
</tr>
<tr>
<td></td>
<td>MOVEKL</td>
<td>Moves the contents of a word into a list specified in the calling sequence</td>
</tr>
<tr>
<td></td>
<td>MOVEAA</td>
<td>Moves whole words from one array to another</td>
</tr>
<tr>
<td></td>
<td>MOVEAL</td>
<td>Moves whole words from an array to a list specified in the calling sequence</td>
</tr>
<tr>
<td></td>
<td>MOVELA</td>
<td>Moves whole words from a list specified in the calling sequence to an array</td>
</tr>
<tr>
<td></td>
<td>MOVELL</td>
<td>Moves one list of words to another</td>
</tr>
<tr>
<td>CAFM</td>
<td>IXTRAC</td>
<td>Selects a character from an array and creates a variable FORTRAN integer</td>
</tr>
<tr>
<td></td>
<td>INSERT</td>
<td>Replaces a character with the low order bits of a FORTRAN integer variable</td>
</tr>
<tr>
<td></td>
<td>MOVECH</td>
<td>Transfers a field of characters from one array to another</td>
</tr>
<tr>
<td></td>
<td>EXTRAC</td>
<td>Selects a character from an array and creates a Hollerith variable</td>
</tr>
<tr>
<td></td>
<td>ENSERT</td>
<td>Places a single Hollerith character into an array</td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th>Subprogram Name</th>
<th>Subroutine Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FORMAT</td>
<td></td>
<td>Formatting</td>
</tr>
<tr>
<td>MOVCMCH</td>
<td></td>
<td>Transfers several fields of characters from one array to another</td>
</tr>
<tr>
<td>MOVECC</td>
<td></td>
<td>Packs or unpacks several fields of characters from one array to another</td>
</tr>
<tr>
<td>MOVEHA</td>
<td></td>
<td>Defines Hollerith constants which may be used as FORTRAN variables</td>
</tr>
<tr>
<td>HEDSET</td>
<td></td>
<td>Sets up to allow automatic writing of headings on reports</td>
</tr>
<tr>
<td>HEDING</td>
<td></td>
<td>Allows the automatic writing of headings on reports</td>
</tr>
<tr>
<td>CLKNDT</td>
<td></td>
<td>Transfers the date and/or time as FORTRAN integers</td>
</tr>
<tr>
<td>DATBCD</td>
<td></td>
<td>Transfers the date as Hollerith characters</td>
</tr>
<tr>
<td>DECMM</td>
<td></td>
<td>Decision Making with FORTRAN Arrays</td>
</tr>
<tr>
<td>CMPAA</td>
<td></td>
<td>Compares logically or arithmetically a number of words of two arrays</td>
</tr>
<tr>
<td>CMPAL</td>
<td></td>
<td>Compares logically or arithmetically a number of words of an array and a list</td>
</tr>
<tr>
<td>CMPPL</td>
<td></td>
<td>Compares logically or arithmetically a number of words of two lists</td>
</tr>
<tr>
<td>CMPKA</td>
<td></td>
<td>Determines if a word exists in a number of words of an array</td>
</tr>
<tr>
<td>CMPKL</td>
<td></td>
<td>Determines if a word exists in a list</td>
</tr>
<tr>
<td>FINDKA</td>
<td></td>
<td>Searches an array for a one word match</td>
</tr>
<tr>
<td>FINDKL</td>
<td></td>
<td>Searches a list for a one word match</td>
</tr>
<tr>
<td>DATAC</td>
<td></td>
<td>Data Conversion</td>
</tr>
<tr>
<td>BCD21</td>
<td></td>
<td>Converts a Hollerith number into a FORTRAN integer</td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th>Subprogram Name</th>
<th>Subroutine Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECIN</td>
<td></td>
<td>Creates a FORTRAN integer from a Hollerith number (two word)</td>
</tr>
<tr>
<td>DECN</td>
<td></td>
<td>Creates a FORTRAN integer from a Hollerith number (one word)</td>
</tr>
<tr>
<td>AAADEC</td>
<td></td>
<td>Creates a signed Hollerith field from a FORTRAN integer</td>
</tr>
<tr>
<td>AADEC</td>
<td></td>
<td>Creates a signed, edited Hollerith field from a FORTRAN integer</td>
</tr>
<tr>
<td>LEDZER</td>
<td></td>
<td>Converts leading of a word to blanks</td>
</tr>
<tr>
<td>RPTGC</td>
<td></td>
<td>Report Generation and Control</td>
</tr>
<tr>
<td>RFTFLT</td>
<td></td>
<td>Real variable report generator control</td>
</tr>
<tr>
<td>RPTFIX</td>
<td></td>
<td>Integer variable report generator control</td>
</tr>
<tr>
<td>IOC</td>
<td></td>
<td>Input/Output Control</td>
</tr>
<tr>
<td>BSFILE</td>
<td></td>
<td>Backspace a file</td>
</tr>
<tr>
<td>UNLOAD</td>
<td></td>
<td>Rewind FORTRAN unit with interlock</td>
</tr>
<tr>
<td>EOFBIN</td>
<td></td>
<td>End-of-file, binary</td>
</tr>
<tr>
<td>EOFFILE</td>
<td></td>
<td>End-of-file, Hollerith</td>
</tr>
<tr>
<td>IOPACK</td>
<td></td>
<td>Blocking and buffering routine for fixed and variable tape records</td>
</tr>
<tr>
<td>COPY</td>
<td></td>
<td>Copy a print file to the printer</td>
</tr>
<tr>
<td>SORTS</td>
<td></td>
<td>Sorting Routines</td>
</tr>
<tr>
<td>BCDSR</td>
<td></td>
<td>Internal commercial and arithmetic sort</td>
</tr>
<tr>
<td>MLSRT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMPAT</td>
<td></td>
<td>Compatibility of EAM/EDP Equipment</td>
</tr>
<tr>
<td>EAM</td>
<td></td>
<td>Creates a word containing the positional representation of the commercial sequence of the characters of a word</td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th>Subprogram Name</th>
<th>Subroutine Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDP</td>
<td></td>
<td>Restores the word of the EAM routine</td>
</tr>
<tr>
<td>CMPEAM</td>
<td></td>
<td>Compares two arrays for commercial sequence</td>
</tr>
<tr>
<td>KMPWL</td>
<td></td>
<td>Compares word fields in two arrays for Field Data sequence</td>
</tr>
<tr>
<td>KMPWA</td>
<td></td>
<td>Compares word fields in two arrays for Algebraic sequence</td>
</tr>
<tr>
<td>KMPCL</td>
<td></td>
<td>Compares character fields in two arrays for Field Data sequence</td>
</tr>
<tr>
<td>KMPCA</td>
<td></td>
<td>Compares character fields in two arrays for Algebraic sequence</td>
</tr>
<tr>
<td>KMPWC</td>
<td></td>
<td>Compares word fields in two arrays for commercial sequence</td>
</tr>
<tr>
<td>KMPWS</td>
<td></td>
<td>Compares word fields in two arrays for scientific sequence</td>
</tr>
<tr>
<td>KMPCC</td>
<td></td>
<td>Compares character fields in two arrays for commercial sequence</td>
</tr>
<tr>
<td>KMPCS</td>
<td></td>
<td>Compares character fields in two arrays for scientific sequence</td>
</tr>
<tr>
<td>LOGSET</td>
<td></td>
<td>Compares (EAM) two strings of Hollerith characters</td>
</tr>
<tr>
<td>JC</td>
<td></td>
<td>Defines a FORTRAN constant</td>
</tr>
<tr>
<td>JD</td>
<td></td>
<td>Divides one FORTRAN constant by another</td>
</tr>
<tr>
<td>JM</td>
<td></td>
<td>Multiplies one FORTRAN constant by another</td>
</tr>
<tr>
<td>ENDFIL</td>
<td></td>
<td>Mark end-of-file on FORTRAN unit</td>
</tr>
<tr>
<td>REWIND</td>
<td></td>
<td>Rewinds FORTRAN unit</td>
</tr>
</tbody>
</table>

**REFERENCE**

*UNIVAC 1107 BEEF Data Processing Manual, (UP-3985), (UNIVAC Division of Sperry Rand Corp., N.Y.).* Copies of this manual are available through the local UNIVAC representative or through the EIN Office at the cost of reproduction and mailing.
SAMPLE INPUT and SAMPLE OUTPUT

Sample Input and Output are included in the reference listed under the USER INSTRUCTIONS. Interested persons are directed to this source.
COST ESTIMATE
See reference listed in USER INSTRUCTIONS.
Charge to user = computer time + postage and handling + network overhead
= computer time + $10.00 + network overhead

CONTENTS—BEFFDP
pages
   1    Identification & Abstract
   3- 6    User Instructions
   7    I/O
   9    Cost—Contents

8/70  $50
DESCRIPTIVE TITLE
BEEF Mathematical Subroutine

CALLING NAME
BEEFM

INSTALLATION NAME
University of Notre Dame
Computation Center

AUTHOR(S) AND AFFILIATION(S)
UNIVAC Division of Sperry Rand Corporation
Westinghouse Electric Corporation
Baltimore Defense and Space Center
Boeing Corporation

LANGUAGE
FORTRAN IV and SLEUTH II

COMPUTER
UNIVAC 1107

PROGRAM AVAILABILITY
Decks and listings presently available

CONTACT
Mrs. Elizabeth Hutcheson, Computer Center, University of Notre Dame,
Notre Dame, Ind. 46556
Tel.: (219) 283-7784

FUNCTIONAL ABSTRACT
The BEEF mathematical library is a set of subroutines supplied by UNIVAC to "enhance FORTRAN's abilities as a scientific processor." This enrichment is in the form of subroutines for the evaluation of mathematical functions, matrix arithmetic, and other standard engineering requirements.

See the User Instructions for further information on BEEFM.

REFERENCE
## USER INSTRUCTIONS

### Program Description

The following subroutines comprise the BEEF Math library.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| DED  | SOLUTION OF A SET OF FIRST ORDER DIFFERENTIAL EQUATIONS  
The subroutines DED, DEI, DES and DEDIS solve a system of N first-order ordinary differential equations having the form  
\[ Y_i' = f_i(t, y_1, y_2, \ldots, y_n) \]  
The user may select one of the following options.  
(a) a fourth order Runge-Kutta method with fixed step size  
(b) Adams-Moulton method with fixed step size  
(c) Adams-Moulton method with variable step size |
| LAGIT | LAGRANGIAN INTERPOLATION  
This subroutine computes \( y = f(x) \) given a discrete set of values of \( x \) and \( y \), using variable degree Lagrangian interpolation by the following formula.  
\[ P_n(x) = \sum_{s=1}^{n} L_s(x)f(x_s) \]  
where  
\[ L_s(x) = \frac{n}{x-t} \frac{(x-x_s)}{(x_t-x_s)} \quad \text{for} \ s = 1, 2, \ldots, n \]  
and  
\[ Y_s = f(x_s) \quad \text{are known for} \ x_s. \] |
| RANDM | UNIFORM OR NORMAL NUMBER GENERATOR  
This subroutine computes either a uniform or normally distributed random number. |
| RKVS  | RUNGE-KUTTA VARIABLE STEP  
This subroutine solves numerically a set of N first-order differential equations (linear or non-linear) using a fifth-order Runge-Kutta method. |
Name | Description
--- | ---
SPLIN | SPLINE INTERPOLATION
This set of subroutines fits a set of points with an interpolating function made up of piecewise curves which act somewhat as an ideal spline. First and second derivatives of the fitted curve are continuous from interval to interval. The user must specify either the first or the second derivatives (but not both) at the end points. The condition that the second derivative be zero at an end means that the end is not held in a bent position.

SETU | GAUSS QUADRATURE
This subroutine provides the abscissa and weight coefficients used in the evaluation of a definite integral by Gauss’s mechanical quadrature formula. This formula differs from Simpson’s Rule and Weddle’s Rule in that the subdivisions are not equidistant but are symmetrically placed with respect to the midpoint of the interval of integration.

COMBES | COMPLEX BESSEL FUNCTION (COMPLEX ARGUMENT AND ORDER)
This subroutine computes the Bessel Functions, $J_{a+n+ib}(x+iy)$ and $Y_{a+n+ib}(x+iy)$ of complex order and argument. The method used is that of Goldstein and Thaler.

NSIMEQ | SIMULTANEOUS LINEAR EQUATION
The subroutine NSIMEQ solves the matrix equation $A \cdot X = B$ for the unknown matrix $X$. The subroutine NDETRM computes a scaled value of the determinant of the A matrix.

ORTHLS | ORTHOGONAL POLYNOMIAL LEAST SQUARES CURVE FITTING
This subroutine finds the polynomial $P_k(x)$ of degree
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CD</td>
<td><em>K</em> which best approximates, in the least squares sense, a weighted set of data points using orthogonal polynomials. The fitted values for a given set of arguments and the fitted value and derivative for a single argument may also be obtained. The user must provide input and output in source program.</td>
</tr>
</tbody>
</table>
| GAMMA      | GAMMA FUNCTION  
This subroutine calculates the value of \( \Gamma(1 + X) \).                                                                                                                                         |
| TABF       | SINGLE INTERPOLATION  
This subroutine performs a *K*-th order interpolation \((K<6)\) on a table of \(X_iY_i\) values giving, for a given value of \(X\), the corresponding value of \(Y\). The method of interpolation used by this subroutine is Aitken's method as described in Ref. 3. Independent variables must be monotonic but need not be equally spaced. |
| BES        | BESSEL FUNCTION (REAL ARGUMENT AND ORDER)  
This subroutine computes the Bessel functions for real argument and order. The method used is that of Goldstein and Thaler.²                                                                 |
| MATAB      | MATRIX ABSTRACTION  
This subroutine performs addition, subtraction, multiplication, postmultiplication and transposition of real and complex matrices.                                                                 |
| MATRTA     | MATRIX ROTATION  
This subroutine will rotate a 3 \( \times \) 3 matrix about any one of the three orthogonal axes through any given angle.                                                                 |
| CINV1      | MATRIX INVERSION AND DETERMINANT EVALUATION OF COMPLEX MATRICES  
This subroutine provides for inversion and determinant evaluation of general complex matrices or real matrices compatible with existing real matrix subroutine. The matrix is inverted by a variation of the Gauss elimination method. |
| INV        | MATRIX INVERSION AND DETERMINANT EVALUATION (REAL)  
This subroutine inverts an \(N \times N\) real matrix or continued
<table>
<thead>
<tr>
<th>Names</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>evaluates its determinant. The mathematical technique employed for matrix is a variation of the Gauss elimination method.</td>
<td></td>
</tr>
<tr>
<td>DETIGN</td>
<td>DETERMINANT AND NORMALIZED EIGENVECTORS OF A COMPLEX MATRIX</td>
</tr>
<tr>
<td>This subroutine calculates the determinant and (if desired) normalized eigenvectors of a complex matrix.</td>
<td></td>
</tr>
<tr>
<td>TBLP</td>
<td>SINGLE LINEAR INTERPOLATION LOOK-UP</td>
</tr>
<tr>
<td>This subroutine performs a single linear interpolation. The interpolation tables must have previously read into the computer. The set of values for the independent variable must be monotonically increasing.</td>
<td></td>
</tr>
<tr>
<td>GDT</td>
<td>SIMULTANEOUS LINEAR EQUATIONS (SINGLE OR DOUBLE PRECISION)</td>
</tr>
<tr>
<td>This subroutine solves a set of N single or double precision simultaneous linear equations.</td>
<td></td>
</tr>
<tr>
<td>INV1</td>
<td>MATRIX INVERSION AND DETERMINANT EVALUATION (SINGLE OR DOUBLE PRECISION)</td>
</tr>
<tr>
<td>This subroutine inverts an N x N real matrix and evaluates its determinant.</td>
<td></td>
</tr>
<tr>
<td>CMPXMD</td>
<td>MULTIPLICATION AND DIVISION OF COMPLEX NUMBER OF LARGE MAGNITUDE</td>
</tr>
<tr>
<td>The subroutines COMMPY and COMDIV perform a complex multiplication and division respectively of large magnitude numbers.</td>
<td></td>
</tr>
<tr>
<td>UDRNRT</td>
<td>UNIFORMLY DISTRIBUTED PSEUDO RANDOM NUMBER GENERATOR</td>
</tr>
<tr>
<td>This subroutine produces a set of uniformly distributed numbers on the unit interval.</td>
<td></td>
</tr>
<tr>
<td>TBLF</td>
<td>PARABOLIC INTERPOLATION</td>
</tr>
<tr>
<td>Given paired values of X and Y, this subprogram calculates the Y value for any particular value of X,XBAR within the range of the given X values.</td>
<td></td>
</tr>
<tr>
<td>PMTR</td>
<td>POLYNOMIAL MATRIX TRIANGULARIZATION ROUTINE</td>
</tr>
<tr>
<td>This subroutine reduces a given polynomial matrix to triangular form. The subroutine saves computer time over the usual method of determinants by treating polynomial elements individually.</td>
<td></td>
</tr>
</tbody>
</table>
ICEM
INTEGRATION WITH CONTROLLED ERROR
This subroutine provides the numerical solution of an N-th order system of linear or non-linear differential equations expressed as a system of N first-order equations.

ARTN
FOUR QUADRANT ARCTANGENT SUBROUTINE
This subroutine computes the arctangent of the ratio of two floating point numbers.

DEFI
DEFINITE INTEGRAL EVALUATION
This subroutine computes the definite integral of functions which are not well behaved. Two estimates of the value of the integral are made, a parabolic approximation, and one using Simpson's Rule. The program will vary the step size to keep the error estimated within specified bounds. In specifying the minimum step size allowed, the user can greatly determine the capabilities of the program in handling oscillating or discontinuous functions.

ERF3
ERROR AND NORMAL FREQUENCY FUNCTION
ERF3 consists of two subroutines, the error function and the normal frequency function. For a given value, Y, the functions are evaluated using approximations for the following integrals

\[ \text{erf}(Y) = \frac{2}{\sqrt{\pi}} \int_{0}^{Y} e^{-t^2} \, dt \approx 1 - \left( \sum_{i=0}^{6} A_i Y^i \right)^{-16} \]

\[ F(Y) = \frac{2}{\sqrt{2\pi}} \int_{0}^{Y} e^{-t^2/2} \, dt \approx \frac{1}{2}[1 + \text{ERF}(Y/\sqrt{2})]. \]

WEGSTN
WEGSTEIN ITERATION SUBROUTINE
Given an implicit equation of the form \( X = f(X) \), the subroutine finds the value for X which provides a specified accuracy in either a relative or absolute sense. The method used is a modification of iteration which determines the root of the explicit function \( g(X) = 0 \) by the iteration process

\[ X_{n+1} = X_n - g(X)/g'(X). \]

\[ f(X_n) - X_n F' \]

\[ X_{n+1} = \frac{1 - F'}{1 - F'} \]

continued
Name Description

BESMEMM  BESSEL FUNCTION EVALUATION
This subroutine will evaluate the Bessel functions for a given floating point argument. Calculation is terminated either when the \( n^{th} \) computed value is equal to the \((n-1)^{th}\) value or when the \( n^{th} \) computed value is less than the absolute value of a pre-assigned test number.

RANNUM  NORMALLY DISTRIBUTED PSEUDO-RANDOM NUMBER GENERATOR
This program will produce a normally-distributed random number with a given mean and standard deviation.

XLAGIN  LAGRANGE INTERPOLATION SUBROUTINE
Given a table of paired values of \( X, Y \), this program computes a value of \( Y \) corresponding to a specific \( XX \) value within the range of the table. The Lagrange interpolation formula is used.

\[
Y = \sum_{k=1}^{n+1} Y_k \prod_{j=1}^{n+1} \frac{XX-X_j}{X_k-X_j} \quad j \neq k
\]

The user must provide input and output routines in a source program. Data for the table need not be in descending order nor equally spaced.

SIM2  DEFINITE INTEGRAL EVALUATION
This subroutine will compute the definite integral of a function using the maximum step size which will provide accuracy within a given bound. The program makes use of Simpson's Rule, computing and comparing two successive terms. Discontinuous functions or functions that oscillate rapidly may result in incorrect results.

DITTO  SPECIALIZED DOUBLE INTEGRAL EVALUATION
This subroutine will evaluate the special definite integral

\[
\int_{yL}^{yu} \int_{xL}^{xu} g(Y) \, dy \, f(X,Y) \, dx
\]

where \( g(Y) \) and \( f(X,Y) \) are defined by function subprograms; \( yu, yL \) and \( xu, xL \) are fixed limits, and \( xL \) is a variable limit equal to \( y \). The user may specify continued
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>the accuracy desired. The basic method of integration used is a variable-interval Simpson's Rule, modified to yield results two orders higher in accuracy than the basic Simpson's Rule.</td>
<td></td>
</tr>
<tr>
<td>FRENEL</td>
<td>FRESNEL INTEGRATION ROUTINE</td>
</tr>
<tr>
<td>This subroutine evaluates the integral</td>
<td></td>
</tr>
<tr>
<td>( (2\pi)^{-1} \int_{0}^{\infty} \frac{e^{-it}}{\sqrt{t}} , dt )</td>
<td></td>
</tr>
<tr>
<td>for any value ( X &gt; 0 ). The output of the subroutine consists of the real and imaginary results of the integration.</td>
<td></td>
</tr>
<tr>
<td>FSERIE</td>
<td>FRACTION SERIES SOLUTION COMPLEX</td>
</tr>
<tr>
<td>This subroutine calculates the ( N ) complex eigenvalues of a complex or real matrix, given ( N+1 ) distinct guesses, ( \lambda_1, \lambda_2, \ldots, \lambda_n ), of the roots and the corresponding ( N+1 ) determinants, ( D(\lambda_1), D(\lambda_2), \ldots, D(\lambda_{n+1}) ). This subroutine may also be used to solve for the roots of the polynomial ( F(x) ). In this case, the guesses ( X_i ) of the roots replace the ( \lambda_i ) and ( F(X_i) ) replaces ( D(\lambda_i) ) for ( i=1,2,\ldots,n+1 ).</td>
<td></td>
</tr>
<tr>
<td>INTEG</td>
<td>SINGLE OR DOUBLE INTEGRAL EVALUATION</td>
</tr>
<tr>
<td>This subroutine will evaluate the double definite integral ( \int_{a}^{b} \int_{c}^{d} f(X,Y) , dy , dx )</td>
<td></td>
</tr>
<tr>
<td>where ( f(X,Y) ) and the limits of integration are defined by the user. The accuracy is also user-specified. The basic method of integration used is a variable-interval Simpson's Rule, modified to yield results two orders higher in accuracy than the basic Simpson's Rule. This program was designed primarily for functions which can be expressed mathematically, rather than tabulated data.</td>
<td></td>
</tr>
<tr>
<td>PLOTXX</td>
<td>PLOT ORDERING SUBROUTINE</td>
</tr>
<tr>
<td>This subroutine will provide a graphical representation of data stored in computer memory for listing on a standard printer. A maximum of 20 curves may be depicted. Scaling is automatic and all titles and plotting symbols may be specified by the user.</td>
<td></td>
</tr>
</tbody>
</table>
Name    Description
ROOTS   ROOTS OF A POLYNOMIAL
This subroutine obtains approximations to the N complex roots of a polynomial with complex coefficients by the Newton-Raphson method. The polynomial must be of degree ≤ 50.

Description of Input
For a complete description of the options available and the usage conventions the user should consult the reference cited below.

Control Cards
Run Card
The Run Card will be provided by University of Notre Dame personnel.

ASG Card
Columns  Contents
1        Multipunch 7 and 8
2-12     ASG A=BEEFM

PLT Card
Columns  Contents
1        Multipunch 7 and 8
2-6      PLT, A
8-72     The user should list the names of the subprograms he desires, separated by commas.

Note: The subprogram name appearing in the left column of the description given in the Functional Abstract should be specified. In this way, all the subroutines comprising a subprogram will be available.

FOR Card
Columns  Contents
1        Multipunch 7 and 8
2-4      FOR

continued
Columns Contents
5 Blank
6–10 User-supplied job name

XQT Card
Columns Contents
1 Multipunch 7 and 8
2–4 XQT
5 Blank
6–10 Job name specified on FOR Card

FIN Card
Columns Contents
1 Multipunch 7 and 8
2–4 FIN

Input Deck
Run Card
ASG Card
PLT Card
FOR Card
[Program Deck]
XQT Card
[Data Cards, if any]
FIN Card

REFERENCES


6/70
SAMPLE INPUT (FOR GAMMA)

- RUN HIDOLL.ZJ1190+3*10
- ASG A=BEFFM
- PLT+A GAMMA
- FOR TEST

**MAIN PROGRAM TO TEST GAMMA FUNCTION ROUTINE**

```
DIMENSION X(50)
PROGRAM INITIALIZATION*
WRITE (6,100)
1 READ (5,101) N
READ (5,102) (X(I),I=1,N)
CALCULATE GAMMA FUNCTION*
DO 2 I=1,N
Y=GMNF(X(I))
OUTPUT*
2 WRITE (6,103) X(I),Y
GO TO 1
```

FORMAT STATEMENTS*

```
100 FORMAT (45HMAIN PROGRAM TO TEST GAMMA FUNCTION ROUTINE.////)
101 FORMAT (110)
102 FORMAT (F10.0)
103 FORMAT (8H FOR X = F15.7*10X,17NGAMMA FUNCTION = F15.7////)
END
```

**XDT TEST**

| 9   | 2.3 |
| 2.0 |
| 1.3 |
| 1.0 |
| 0.0 |
| 0.7 |
| 1.7 |
| 2.7 |

**FIN**
SAMPLE OUTPUT (FROM GAMMA)

MAIN PROGRAM TO TEST GAMMA FUNCTION ROUTINE.

<table>
<thead>
<tr>
<th>FOR Y =</th>
<th>GAMMA FUNCTION =</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.000000</td>
<td>2.41886796166</td>
</tr>
<tr>
<td>2.000000</td>
<td>2.00000000000</td>
</tr>
<tr>
<td>1.000000</td>
<td>1.00000000000</td>
</tr>
<tr>
<td>1.000000</td>
<td>1.00000000000</td>
</tr>
<tr>
<td>0.000000</td>
<td>1.00000000000</td>
</tr>
<tr>
<td>-0.000000</td>
<td>2.40157466</td>
</tr>
<tr>
<td>-0.000000</td>
<td>2.40157466</td>
</tr>
<tr>
<td>-0.000000</td>
<td>2.40157466</td>
</tr>
</tbody>
</table>

ATTEMPT TO TAKE GAMMA FUNCTION OF -0.000000001
SAMPLF INPUT (FOR BES)

- RUN MIDOLL
- AEG 4=RESTFM
- PLT A=REST

FOR TEST
- MAIN PROGRAM TO TEST BESSEL FUNCTION SUBROUTINE FOR REAL ARGUMENT AND ORDER.

DIMENSION BJ(250), BY(250), BI(250), BK(250), TX(201), TFNU(20), IN(10)

PROGRAM INITIALIZATION.

WRITE (6,100)
1 READ (5,110) NX, NNU, NON
READ (5,111) (TF(I), I=1,NX)
READ (5,111) (TFNU(I), I=1,NU)
READ (5,110) (IN(I), I=1,NON)

LOOP TO TAKE ALL POSSIBLE VALUES, ONE FROM EACH TABLE.

DO 10 IM1 = 1,NX
  DO 10 J = 1,NU
  DO 10 K = 1,NON
  XM TX(I)
  FNU = TFNU(J)
  N = IN(K)
  N1 = IAES(N)+1

CALL IP 3 SEQUENCE.

CALL EESY(X,FNU,N,BI,BK)
GO TO 2
3 CALL ESK (X,FNU,N,BI,BK)
GO TO 4

OUTPUT.

2 WRITE (6,102) X, FNU, N
WRITE (6,103) (BI(L), L=1,N1)
WRITE (6,104) (BY(L), L=1,N1)
GO TO 3
4 WRITE (6,105) (BI(L), L=1,N1)
WRITE (6,106) (BK(L), L=1,N1)
10 CONTINUE

FORMAT STATEMENTS.

100 FORMAT (77HMAIN PROGRAM TO TEST BESSEL FUNCTION SUBROUTINE FOR REAL ARGUMENT AND ORDER.///)
102 FORMAT (5X2H=EI4.7,5X3HUM=EI4.7,5X2HN=15///)
103 FORMAT (17HMOOUTPUT FROM BESI///SE17.8)!!)
104 FORMAT (17HMOOUTPUT FROM BESY///SE17.8)!!)
105 FORMAT (17HMOOUTPUT FROM BESK///SE17.8)!!)
106 FORMAT (14H15.0)

END

NOT TEST
8 1 2
3 4 5
6 2.404826 3.957678 7.015987
7 9 -9

FIN
SAMPLE OUTPUT (FROM BES)

"AIX PROP" TO TEST PESEL FRACTION COMPUTING FOR ALL ARGUMENT AND ORDER.

v = 0.0000000001

\[ \begin{align*}
\text{OUTPUT FOR PESEL} & \quad v = 0.0000000001 \\
-0.260519 & -0.260519 \\
-0.33 & -0.33 \\
\text{OUTPUT FOR PESJ} & \quad v = 0.0000000001 \\
-0.260519 & -0.260519 \\
-0.33 & -0.33 \\
\text{OUTPUT FOR PESE} & \quad v = 0.0000000001 \\
-0.260519 & -0.260519 \\
-0.33 & -0.33 \\
\text{OUTPUT FOR PESI} & \quad v = 0.0000000001 \\
-0.260519 & -0.260519 \\
-0.33 & -0.33 \\
\end{align*} \]
<table>
<thead>
<tr>
<th>Reference</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>4550mC</td>
<td>4.10448</td>
</tr>
<tr>
<td>4364mC</td>
<td>4.10448</td>
</tr>
<tr>
<td>3597mC</td>
<td>4.10448</td>
</tr>
<tr>
<td>2854mC</td>
<td>4.10448</td>
</tr>
<tr>
<td>2167mC</td>
<td>4.10448</td>
</tr>
<tr>
<td>1745mC</td>
<td>4.10448</td>
</tr>
<tr>
<td>1420mC</td>
<td>4.10448</td>
</tr>
</tbody>
</table>

continued
SAMPLE INPUT (FOR MATAB)

Test programs for real matrix addition and subtraction.

- RUN MIDOLL.ZJ1190.310
- ASG A=BEFFM
- PLT A MATAB

FOR TESTS

MAIN PROGRAM TO TEST 1107 MATRIX PACKAGE - REAL MATRIX ADDITION
AND SUBTRACTION. SET J=1 FOR ADDITION J=0 FOR SUBTRACTION.

DIMENSION A(64),8(64),C(64)

PROGRAM INITIALIZATION.

WRITE (6,100)
1 READ (5,101) J,L,M,N
N1 = L*M
READ (5,102) (A(I),I=1,N1)
READ (5,102) (B(I),I=1,N1)

IF (J.EQ.0) GO TO 2
CALL P.XA(A,B,C,M,N)
GO TO 3

2 CALL P.XS(A,B,C,M,N)
GO TO 4

OUTPUT

3 WRITE (6,103) (C(I),I=1,N1)
GO TO 1

4 WRITE (6,104) (C(I),I=1,N1)

FORMAT STATEMENTS.

100 FORMAT (59HM) PROGRAM TO TEST REAL MATRIX ADDITION AND SUBTRACT
110N+///
101 FORMAT (4I5)
102 FORMAT (F10.0)
103 FORMAT (/69H MATRIX C FOR REAL ADDITION ROUTINE:///1M E20.8))
104 FORMAT (/69H MATRIX C FOR REAL SUBTRACTION ROUTINE:///1M E20.8))

END

NOT TEST

1 3 2 3 4 1 4 1 2 1 4 1 2 1 0 
5 3 2 3 4 1 4 1 2 1 0 3 2 3 4 1 4 1 2 1
SAMPLE OUTPUT (FROM MATLAB)

Test program for real matrix addition and subtraction.

"MAIN PROGRAM TO TEST REAL MATRIX ADDITION AND SUBTRACTION."

"MATRIX C FOR REAL ADDITION ROUTINE."

```
1.00000000+01
1.00000000+01
1.00000000+01
1.00000000+01
1.00000000+02
1.00000000+01
1.00000000+01
1.00000000+01
```

"MATRIX C FOR REAL SUBTRACTION ROUTINE."

```
1.00000000+01
1.00000000+01
1.00000000+01
1.00000000+01
1.00000000+01
1.00000000+01
1.00000000+01
1.00000000+01
```

C
Test program for real matrix multiplication.

```plaintext
DIMENSION A(3*3), B(3*3), C(3*3)

WRITE (6,100)
1 READ (5,101) M, N, L
READ (5,102) (A(I,J), J=1,L), I=1,M
READ (5,102) (B(I,J), J=1,N), I=1,M
CALL MXMIA, B, C, M, N, L

WRITE (6,103) (C(I,J), I=1,M, J=1,N)
GO TO 1

100 FORMAT(9HMAIN PROGRAM TO TEST REAL MATRIX MULTIPLICATION.../)
101 FORMAT(3I6)
102 FORMAT(F12.6)
103 FORMAT(4X2H1*3X2HJ*8X7H C(I,J)/(11H 215E20.8))

FIN
```

XQTTEST4

```
3 3 3
 3
 2
 2
 3
 4
 3
 0
 0
 5
 5
 1
 1
 1
 5
 1
 1
 1
 1
 3
 3
 3
 3
 2
 2
 2
 4
 3
 0
 0
 5
 5
 2
 2
 2
 2
 2
 2

FIN
```

6/70
25
SAMPLE OUTPUT (FOR MATLAB)

Test program for real matrix multiplication.

```
% MAIN PROGRAM TO TEST REAL MATRIX MULTIPLICATION.

for i = 1:3
    for j = 1:3
        C(i,j) = A(i,:)*B(:,j);
    end
end
```

Execution terminated by an attempt to read thru ENDOFPDATA.
Above fault at internal sequence number 00145 in TESTP4.
SAMPLE INPUT (FOR MATLAB)
Test program for complex matrix postmultiplication.

- Run HIDOLL2J1190*3*10
- ASG A=BEFFM
- FOR TEST
- MAIN PROGRAM TO TEST 1107 MATRIX PACKAGE - COMPLEX MATRIX POSTMULTIPLICATION.
DIMENSION A(2,3*3), B(2,3*3) C(2,3*3)

PROGRAM INITIALIZATION.
WRITE (6,100)
READ (5,101) M, N
READ (5,102) ((A(I,J)+A(2,J),J=1,N),I=1,M)
READ (5,102) ((B(I,J)+B(2,J),J=1,N),I=1,M)
CALL (MXMD1A,B,C,M,N)
OUTPUT.
WRITE (6,103) ((C(I,J),J=1,N),I=1,M)
GO TO 1

FORMAT STATEMENTS.
100 FORMAT (56H MAIN PROGRAM TO TEST COMPLEX MATRIX POSTMULTIPLICATION.
110 (I//)
120 FORMAT (215)
121 FORMAT (2F10.0)
122 FORMAT (25H POSTMULTIPLICATION OF A, OUTPUT FROM 1107 MATRIX PACKAGE - COMPLEX MATRIX POSTMULTIPLICATION.
1100 1107 MATRIX PACKAGE - COMPLEX MATRIX POSTMULTIPLICATION.
1110 END

- XOT TEST
3 3
2 1 1
2 1 1
3 1 1
4 1 1
5 1 1
6 1 1
7 1 1
8 1 1
9 1 1
2 1 1
2 1 1
2 1 1
2 1 1

- FIN
SAMPLE OUTPUT (FROM MATLAB)

Test program for complex matrix postmultiplication.

"MAIN PROGRAM TO TEST COMPLEX MATRIX POSTMULTIPLICATION.

POSTMULTIPLICATION OF A.

<table>
<thead>
<tr>
<th>i</th>
<th>j</th>
<th>A(i,j)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>3000000000+01</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3000000000+01</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3000000000+01</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3000000000+01</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1100000000+01</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1100000000+01</td>
</tr>
</tbody>
</table>

POSTMULTIPLICATION OF A.

<table>
<thead>
<tr>
<th>i</th>
<th>j</th>
<th>A(i,j)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>-60000000</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>-60000000</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>-60000000</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>-60000000</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>-60000000</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>-60000000</td>
</tr>
</tbody>
</table>

EXECUTION TERMINATED BY AN ATTEMPT TO READ THIN END-OF-DATA.

AFTER FAULT AT INTERNAL SEQUENCE NUMBER 0014 IN PS.
SAMPLE INPUT (FOR MATAB)
Test program for complex matrix transpose.

RUN HIOOOL *ZJ1190.3+10
ASG A=REF4
PLT A MATI3
FOR TEST
MAIN PROGRAM TO TEST 1107 MATRIX PACKAGE - COMPLEX MATRIX TRANSPOS
DIMENSION A(2,5,3), C(2,3,5)
PROGRAM INITIALIZATION*
WRITE (6,100)
1 READ (5,101) M, N
READ (5,102) ((A(1,1,J)+A(2,1,J)+J=1,N)+I=1,M)
CALL CMXT(A+C+M+N)
OUTPUT*
WRITE (6,103) ((1,1,J)+C(1,1,J)+C(2,1,J)+J=1,N)+I=1,M)
GO TO 1
FORMAT STATEMENTS*
100 FORMAT (47AMAIN PROGRAM TO TEST COMPLEX MATRIX TRANSPOSE*///)
101 FORMAT (215)
102 FORMAT (2F10.0)
103 FORMAT (31M COMPLEX MATRIX TRANSPOSE OF A*///4X2M I,3X2M J, 19X7M A
1(1,1,J)///1M 715 2E20 E8))
END
XOT TEST
5 3
6* 0*
4* -2*
1* 0*
0* 10*
6* -2*
4* -4*
2* -8*
8* 9*
4* -2*
3* 2*
4* 6*
2* -8*
3* -2*
2* -1*
1* 0*
FIN
SAMPLE OUTPUT (FROM MATAB)
Test program for complex matrix transpose.

"MAIN PROGRAM TO TEST COMPLEX MATRIX TRAPOSE."

"COMPLEX MATRIX TRAPOSE OF A."

<table>
<thead>
<tr>
<th>T</th>
<th>J</th>
<th>A(T+J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>+8000000001+02</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>+8000000001</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>+8000000001</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>+8000000001</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>+8000000001+01</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>+8000000001+01</td>
</tr>
</tbody>
</table>

EXECUTION TERMINATED BY AN ATTEMPT TO READ THE END-OF-DATA.
ABOVE FAULT AT INTERNAL SEQUENCE NUMBER 0019 IN TEST.
SAMPLE INPUT (FOR CINV1)

- RUN HIDOLL, ZJ119O, 3.10
- ASG =REFFM
- PLT= INV>CINV1
- FOR TEST?
  MAIN PROGRAM TO TEST DETERMINANT EVALUATION AND MATRIX INVERSION
  OF COMPLEX AND REAL MATRICES. SET IER=0 FOR COMPLEX MATRICE; IER=1
  FOR REAL MATRICES. DIVIDE CHECK USED FOR SINGULARITY.
  DIMENSION AREAL(50), BIMAG(50), K(1), IK(1)
  EQUIVALENCE (AREAL(1), K(1)), (AREAL(2), K(2)), (AREAL(3), K(3)), (BIMAG
  1(1), IK(1)), (BIMAG(2), K(2)), (BIMAG(3), IK(3))

  PROGRAM INITIALIZATION.

  WRITE (6, 100)
  1 READ (5, 101) N, IER
  IF (IER.EQ.0) GO TO 6
  READ (5, 102) (AREAL(L), L = M + 1, M)
  K(2) = L
  K(3) = L
  GO TO 5

  2 READ (5, 103) (AREAL(L), BIMAG(L), L = M + 1, M)
  IK(2) = L
  IK(3) = L

  CALL SEQUENCE.

  CALL CINV1 (AREAL, BIMAG, IER)
  IF (IER.EQ.0) GO TO 7
  IF (IER.EQ.-1) GO TO 8
  GO TO 10

  5 DO 6, 1 = M
  BIMAG(L) = 0.0
  6 CONTINUE

  CALL CINV1 (AREAL, BIMAG, IER)
  IF (IER.EQ.0) GO TO 7
  IF (IER.EQ.-1) GO TO 8
  GO TO 11

  7 WRITE (6, 104)
  GO TO 1

  8 WRITE (6, 105)
  GO TO 1

  10 WRITE (6, 106) AREAL(1), BIMAG(1)
  WRITE (6, 107) (AREAL(L), BIMAG(L), L = M + 1, M)
  GO TO 1

  11 WRITE (6, 108) (AREAL(L), L = M + 1, M)
  WRITE (6, 109) (AREAL(L), L = M + 1, M)
  GO TO 1

  PRINT STATEMENTS.

  100 FORMAT (96H MAIN PROGRAM TO TEST INVERSION AND DETERMINANT EVALUATION
  OF GENERAL COMPLEX AND REAL MATRICES.///)

  101 FORMAT (2I5)
  102 FORMAT (F10.0)
  103 FORMAT (F10.0,F10.0)

continued
104 FORMAT(20H MATRIX IS SINGULAR.///)
104 FORMAT(31H MATRIX NOT IN ACCEPTABLE FORM.///)
106 FORMAT(15H DETERMINANT = 2E20.8///)
107 FORMAT(16H INVERSE MATRIX///11H 2E20.8))
108 FORMAT(15H DETERMINANT = E20.8///)
109 FORMAT(16H INVERSE MATRIX///11H E20.8))
END

- XOT TEST2
  3 1
  9.3746 3.0416 2.4371 3.0416 6.1832 1.2163 2.4371 1.2163 8.4419 3.0
  2.4371 -2.4371 1.2163 1.2163 3.4429 3.4429
- FIN
SAMPLE OUTPUT (FROM CINV1)

"MAIN PROGRAM TO TEST INVERSION AND DETERMINANT EVALUATION OF GENERAL COMPLEX AND REAL MATRICES.

DETERMINANT = .3526593843

INVERSE MATRIX:

.16873262-00
-.64943731-01
-.43565076-00
-.50937609-01
-.62932875-01
-.45036700-01
-.16219389-00

DETERMINANT = -.1018148497 1.0319158849

INVERSE MATRIX:

.69992537-01 1.06827532-01
-.67777260-01 .42773922-01
-.48777240-01 .67777240-01
-.19390010-00 .17298000-00
-.6292895-01 .8942925-01
-.8739292-01 .64943731-01
-.4299232-01 .50937609-01
-.2650333-00 .3299039-00

EXECUTION TERMINATED BY AN ATTEMPT TO READ THRU PND-OF-DATA.
ABOVE FAULT AT INTERNAL REGU-0E SEP 000 IN TP33.
COST ESTIMATE

For the jobs listed on the Sample Input, the processing times are listed below. The current rate for the University of Notre Dame is $480./hour for computer time with an additional charge of $10. for postage and handling.

Charge to user = computer costs + postage and handling + network overhead

<table>
<thead>
<tr>
<th>Program</th>
<th>Processing Time (in seconds)</th>
<th>Computer costs</th>
<th>Charge to user</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAMMA</td>
<td>16</td>
<td>$2.51</td>
<td>$12.51</td>
</tr>
<tr>
<td>BES</td>
<td>23</td>
<td>$3.74</td>
<td>$13.74</td>
</tr>
<tr>
<td>MATAB (add. &amp; sub.)</td>
<td>32</td>
<td>$4.67</td>
<td>$14.67</td>
</tr>
<tr>
<td>MATAB (multiplic.)</td>
<td>43</td>
<td>$6.11</td>
<td>$16.11</td>
</tr>
<tr>
<td>MATAB (postmul ip.)</td>
<td>29</td>
<td>$4.23</td>
<td>$14.23</td>
</tr>
<tr>
<td>MATAB (transpose)</td>
<td>32</td>
<td>$4.74</td>
<td>$14.74</td>
</tr>
<tr>
<td>CINV1</td>
<td>32</td>
<td>$4.82</td>
<td>$14.82</td>
</tr>
</tbody>
</table>

* A figure for network overhead must still be added to the totals in this column.

CONTENTS—BEEFM

pages
1 Identification & Abstract
3-11 User Instructions
13-14 I/O (GAMMA)
15-20 I/O (BES)
21-30 I/O (MATAB)
31-33 I/O (CINV1)
35 Cost—Contents
DESCRIPTIVE TITLE
Bio-Medical Multivariate Statistical Programs

CALLING NAME
BMD

INSTALLATION NAME
University of Notre Dame Computing Center

AUTHOR(S) AND AFFILIATION(S)
School of Medicine
University of California, Los Angeles
UNIVAC Division of Sperry Rand Corporation
College of Business Administration and Computing Center
University of Notre Dame

LANGUAGE
FORTRAN IV

COMPUTER
UNIVAC 1107

PROGRAM AVAILABILITY
Decks and listings presently available

CONTACT
Elizabeth Hutcheson, EIN Technical Representative, Computing Center,
University of Notre Dame, Notre Dame, Ind. 46556
Tel.: (219) 283-7784

FUNCTIONAL ABSTRACT
The BMD system is a package of computer programs designed to do both basic data processing and the subsequent statistical analysis. The programs have been prepared in an easy-to-use parametric form so that the researcher may adapt them to a wide variety of statistical problems. For further details of the package, see the User Instructions. The BMD is available at the University of Notre Dame in its 1967 edition.

REFERENCES

Univ. of Notre Dame Computing Center, BMD for the UNIVAC 1107, (Rough draft, 1967 ed. of BMD). Available from Univ. of Notre Dame Computing Center, Notre Dame, Ind.
USER INSTRUCTIONS

The computer routines which constitute the BMD multivariate statistical package are designed for use by statisticians and researchers; their use does not require the services of a computer programmer. The user of the statistical system enters his data on punched cards in an easy-to-use and flexible format. A problem header card is also punched which defines the size and options to be used for each analysis to be performed. Detailed instructions for the preparation of these cards is included in Ref. 1 and 2.

The card decks are then forwarded to the UNIVAC 1107 or 1108 computer installation where system cards are added. These systems cards call the appropriate computer program and define any tape files used. The analysis is then performed on the computer and the print-outs sent back to the researcher. Many computer installations are able to perform this service (even for large computations) for researchers within a day.

Keypunching of the basic input can, however, often be a detailed and time consuming operation. Researchers should contact the keypunching department before they start accumulating their raw data so that it can be collected in a format lending itself to rapid, accurate transcription.

The following programs comprise the BMD package:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMD01D</td>
<td>SIMPLE DATA DESCRIPTION</td>
</tr>
<tr>
<td></td>
<td>This program computes simple averages and measures of dispersion of variables, omitting those values which the user specifies for exclusion from the computations.</td>
</tr>
<tr>
<td>BMD02D</td>
<td>CORRELATION WITH TRANSGENERATION</td>
</tr>
<tr>
<td></td>
<td>This program computes simple correlation coefficients, averages and measures of dispersion on entering variables and/or transgenerated variables from selected cases whose values for specified variables have a precise logical relationship in agreement with a specified Boolean expression. Cross-tabulation plots are optional.</td>
</tr>
<tr>
<td>BMD03D</td>
<td>CORRELATION WITH ITEM DELETION</td>
</tr>
<tr>
<td></td>
<td>This program computes a simple correlation matrix,</td>
</tr>
</tbody>
</table>

Continued
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMD04D</td>
<td><strong>ALPHANUMERIC FREQUENCY COUNT (1 COLUMN DATA)</strong>&lt;br&gt;This program computes frequencies of legal characters on one-column data. Any legal numeric, alphabetic, or special character is counted. The input data may be on cards, or on magnetic tape in Field Data mode.</td>
</tr>
<tr>
<td>BMD05D</td>
<td><strong>GENERAL PLOT INCLUDING HISTOGRAMS</strong>&lt;br&gt;This program provides a method by which graphs and histograms can be produced.</td>
</tr>
<tr>
<td>BMD06D</td>
<td><strong>DESCRIPTION OF STRATA</strong>&lt;br&gt;Cases are separated into groups based on specified intervals of one variable, the conditioning variable. For these selected groups, computations are performed for any other variables which are designated as the conditioned variables. Univariate descriptions and correlation coefficients are included.</td>
</tr>
<tr>
<td>BMD07D</td>
<td><strong>DESCRIPTION OF STRATA WITH HISTOGRAMS</strong>&lt;br&gt;This program groups the data into a specified number of groups based on the order of entry of the data or into groups whose values for a base variable are within intervals established by specified cut points. For these groups, histograms are printed for each variable. The number of classes or categories of the histograms may be specified or they may be computed by the program. Univariate descriptions and correlation coefficients are included.</td>
</tr>
<tr>
<td>BMD08D</td>
<td><strong>CROSS TABULATION WITH VARIABLE STACKING</strong>&lt;br&gt;This program computes two-way frequency tables of data input. Frequency tables are computed from specified ranges of the original variables, variables after transgeneration, stacked variables or combinations of these. Data input may be positive or negative integers only.</td>
</tr>
</tbody>
</table>
| BMD09D   | **CROSS TABULATION, INCOMPLETE DATA**<br>This program performs cross-tabulations of input data, continued...
Name | Description
--- | ---
excluding from cross-tabulation specified special values or codes used to designate missing values. The program differentiates between blanks and zeros in testing for special values.
BMD10D | DATA PATTERNS FOR DICHOTOMIES
This program finds frequencies and patterns of any one particular specified code in the input data. Frequent use will be for a code representing missing values.
BMD11D | DATA PATTERNS FOR POLYCHOTOMIES
This program prints patterns of one-column data and item numbers or case numbers to identify cases having these data patterns. If desired, original data may be recoded by the program before the data patterns are printed.
BMD01M | PRINCIPAL COMPONENT ANALYSIS
This program computes the principal components of standardized data and rank-orders each standardized case by the size of each principal component separately.
BMD02M | REGRESSION ON PRINCIPAL COMPONENTS
This program computes the principal components of standardized data and rank-orders each standardized case by the size of each principal component separately. Each dependent variable is regressed on the first, first two, first three and all principal components when each component is expressed in terms of standardized data.
BMD03M | GENERAL FACTOR ANALYSIS
This program performs a principal component solution and an orthogonal rotation of the factor matrix. Data input to this program may be in the form of raw data, a correlation matrix, or a factor matrix. Data input may be read in from punched cards, Field Data or binary tape.
BMD04M | DISCRIMINANT ANALYSIS FOR TWO GROUPS
This program computes a linear function of p variables measured on each individual of two groups. This function can serve as an index for discrimination between the groups. It is determined from
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>from the criterion of &quot;best&quot; in that the difference between the mean indices for the two groups divided by a pooled standard deviation of the indices is maximized.</td>
<td></td>
</tr>
</tbody>
</table>
| BMD05M             | DISCRIMINANT ANALYSIS FOR SEVERAL GROUPS  
This program directs the computation of a set of linear functions for the purpose of classifying an individual into one of several groups. The input data consist of a set of observations for each of the classification groups; each observation consists of the values of a set of variables, and each observation contains a value for each of the variables. |
| BMD06M             | CANONICAL ANALYSIS  
This program computes the canonical correlations between two sets of variables. |
| BMD07M             | STEPWISE DISCRIMINANT ANALYSIS  
This program performs a multiple discriminant analysis in a stepwise manner. At each step one variable is entered into the set of discriminating variables. The entering variable is selected by the following equivalent criteria; the variable with the largest F value (see computational procedure), or the variable which when partialled on the previously entered variables has the highest multiple correlation with the groups, or the variable which gives the greatest decrease in the ratio of within to total generalized variances. A variable is deleted if its F value becomes too low. The program also computes canonical correlations and coefficients for canonical variables. It plots the first two canonical variables to give an optimal two-dimensional picture of the dispersion. |
| BMD01R             | SIMPLE LINEAR REGRESSION  
This program performs simple linear regression analysis on a single or combined treatment groups with unequal sample sizes. (The words "treatments groups" are used here to describe categories.) The "within" cross-products sums and coefficients are computed; thus, analysis-of-covariance information is also provided in the output. |
Names | Description
--- | ---
BMD02R | STEPWISE REGRESSION
This program computes a sequence of multiple linear regression equations in a stepwise manner. At each step one variable is added to the regression equation. The variable added is the one which makes the greatest reduction in the error sum of squares. Equivalently, it is the variable which has highest partial correlation with the dependent variable partialed on the variables which have already been added; and equivalently, it is the variable which, if it were added, would have the highest F value. In addition, variables can be forced into the regression equation. Non-forced variables are automatically removed when their F values become too low. Regression equations with or without the regression intercept may be selected.

BMD03R | MULTIPLE REGRESSION WITH CASE COMBINATIONS
This program performs multiple regression and correlation analysis on the data within various combinations of subsamples from the same population.

BMD04R | PERIODIC REGRESSION AND HARMONIC ANALYSIS
This program performs periodic or harmonic regression analysis using the regression function of the Fourier series form. Periodic regressions are computed successively up to the harmonic specified by the user.

BMD05R | POLYNOMIAL REGRESSION
This program computes polynomial regressions of the form

\[ Y = \alpha + \beta_1 X + \beta_2 X^2 + \cdots + \beta_k X^k \] (kth degree),

where \( k \) is some positive integer.

BMD06R | ASYMPTOTIC REGRESSION
This program performs asymptotic regression analysis using the modified exponential function of the form,

\[ Y = \alpha + \beta_x x. \]

By making a transgeneration (or successive transgenerations) on the dependent variable \( Y \), other equations can be used.

continued
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMD01S</td>
<td>LIFE TABLE AND SURVIVAL RATE</td>
</tr>
<tr>
<td></td>
<td>This program computes proportions surviving, survival rates and standard</td>
</tr>
<tr>
<td></td>
<td>errors for successively reduced time periods and prints a plot of the</td>
</tr>
<tr>
<td></td>
<td>cumulative proportion surviving.</td>
</tr>
<tr>
<td>BMD02S</td>
<td>CONTINGENCY TABLE ANALYSIS</td>
</tr>
<tr>
<td></td>
<td>This program computes two-way frequency and percentage tables, chi squares,</td>
</tr>
<tr>
<td></td>
<td>contingency coefficients, and maximum likelihood ratios. Each variable may</td>
</tr>
<tr>
<td></td>
<td>be categorized in several different ways; each way is referred to as a</td>
</tr>
<tr>
<td></td>
<td>categorization.</td>
</tr>
<tr>
<td>BMD03S</td>
<td>BIOLOGICAL ASSAY; PROBIT ANALYSIS</td>
</tr>
<tr>
<td></td>
<td>This program obtains maximum likelihood estimates for the parameters $\alpha$</td>
</tr>
<tr>
<td></td>
<td>and $\beta$ in the probit equation $y = \alpha + \beta x$. An iterative</td>
</tr>
<tr>
<td></td>
<td>scheme is used. As an option, an estimate of the natural (threshold)</td>
</tr>
<tr>
<td></td>
<td>response rate is also obtained.</td>
</tr>
<tr>
<td>BMD04S</td>
<td>GUTTMAN SCALE PREPROCESSOR</td>
</tr>
<tr>
<td></td>
<td>This program performs the initial step of BMD06S and BMD07S (GUTTMAN SCALE</td>
</tr>
<tr>
<td></td>
<td>NO. 2, PARTS 1 AND 2). It weights the input data, checks the data,</td>
</tr>
<tr>
<td></td>
<td>determines the frequencies of responses, combines those responses which</td>
</tr>
<tr>
<td></td>
<td>have frequencies less than N% (N is user-specified) of the total</td>
</tr>
<tr>
<td></td>
<td>respondents, ranks the respondents by the Cornell technique, and</td>
</tr>
<tr>
<td></td>
<td>determines the errors.</td>
</tr>
<tr>
<td>BMD05S</td>
<td>GUTTMAN SCALE NO. 1</td>
</tr>
<tr>
<td></td>
<td>This program takes the data on many variables, assigns proper weights to</td>
</tr>
<tr>
<td></td>
<td>the data, ranks the respondents (or cases) from most favorable to least</td>
</tr>
<tr>
<td></td>
<td>favorable, and assigns a Guttman scale score for each respondent. This</td>
</tr>
<tr>
<td></td>
<td>procedure is widely used for data from attitude or opinion questionnaires.</td>
</tr>
<tr>
<td>BMD06S</td>
<td>GUTTMAN SCALE NO. 2, PART 1</td>
</tr>
<tr>
<td></td>
<td>This program performs the initial steps for GUTTMAN SCALE NO. 2. The</td>
</tr>
<tr>
<td></td>
<td>Problem, Response, and Reflection (if any) Cards are read and processed.</td>
</tr>
<tr>
<td></td>
<td>The input data are read into the computer, weighted, and checked for valid</td>
</tr>
<tr>
<td></td>
<td>values.</td>
</tr>
</tbody>
</table>

continued
**Name**  
**Description**  

**BMD07S**  
GUTTMAN SCALE NO. 2, PART 2  
This program performs the major computations of GUTTMAN SCALE NO. 2. It reads and processes any Forced Combination Cards and performs any and all combinations of responses which will give the best possible Guttman scale the program can produce. During the running of the program, the respondents are ranked by the Cornell technique after each combination and the errors to each response are determined.

**BMD08S**  
GUTTMAN SCALE NO. 2, PART 3  
This program performs the final steps of GUTTMAN SCALE NO. 2. In this program, weights are assigned to all zero responses such that no additional errors are introduced. Guttman scale scores are assigned, and the final results specified by the user are printed out.

**BMD09S**  
TRANSCATION  
This program performs selected transgenerations on specified variables in the data. Any of the trans-generation codes listed in the BMD manual may be selected. Input may be from punched cards, from Field Data tape, or from binary tape.

**BMD10S**  
TRANSPOSTION OF LARGE MATRICES  
This program performs no analysis but is designed to convert large matrices to the form required by some programs which do perform statistical analysis (e.g., to convert data from "case-wise" to "variable-wise" form for use in BMD01V). Large matrices are read in from cards or Field Data tape, transposed, and written out on Field Data tape.

**BMD01T**  
AMPLITUDE AND PHASE ANALYSIS  
This program computes the amplitude and phase of moderately wide-band noise and noise contaminated by extraneous noise. The amplitude and phase are determined from a pair of finite moving averages on the sample noise. A generalized Tukey filter with a variable number of triangles and resolutions is used.

*continued*
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| BMD02T   | AUTOCOVARIANCE AND POWER SPECTRAL ANALYSIS  
This program computes the autocovariance, power spectrum, cross covariance, cross spectrum, transfer function, and coherence function of time series. |
| BMD01V   | ANALYSIS OF VARIANCE FOR ONE-WAY DESIGN  
This program computes an analysis-of-variance table for one variable of classification, with unequal group sample sizes. |
| BMD02V   | ANALYSIS OF VARIANCE FOR FACTORIAL DESIGN  
This program computes an analysis of variance for a factorial design. |
| BMD03V   | ANALYSIS OF COVARIANCE FOR FACTORIAL DESIGN  
This program computes a full factorial analysis of covariance. |
| BMD04V   | ANALYSIS OF COVARIANCE WITH MULTIPLE COVARIATES  
This program is designed to compute analysis-of-covariance information for one analysis-of-variance variable with multiple covariates and unequal treatment group sizes. Cases may be specified by the user as being in certain treatment groups, or cases may be placed in treatment groups by the program in accordance with a specified Boolean expression. |
| BMD05V   | GENERAL LINEAR HYPOTHESIS  
This program performs the calculations required for a general linear hypothesis model. The independent variables are of two general types, variables used to specify the analysis-of-variance classification, and variables used as covariates. By use of these variables, the program can be used for balanced or unbalanced analysis-of-variance or covariance designs and missing-value problems. |
| BMD06V   | GENERAL LINEAR HYPOTHESIS WITH CONTRASTS  
This program is similar to BMD05V in that it is designed to estimate and test the statistical significance of the parameters which occur in the general linear hypothesis model. This program is more general in that it can test the statistical significance of any real valued linear function |
Name  Description

of the parameters. It is less general in that
higher order simultaneous tests cannot be made.
Thus, this program can test the hypothesis $\beta_1 = \beta_2 = 0$.

**BMD07V**  MULTIPLE RANGE TESTS
This program computes an analysis-of-variance table
for one variable of classification, with unequal
group sizes. The treatment group or category means
are ranked in increasing order; and a multiple range
test, using significant ranges input by the user,
is performed on the ranked means. A maximum of 9
range tests may be done on the ranked means.

**BMD08V**  ANALYSIS OF VARIANCE
This program performs analysis of variance for any
hierarchial design with equal cell sizes. This
includes the nested, partially nested and partially
crossed, and fully crossed designs. The model is
specified by indicating the nesting relationship
of the indices. Separate analyses may be performed
on several dependent variables simultaneously. Each
analysis of variance table includes an expected mean
square in terms of the population variance components.
For this calculation each index may be specified as
fixed or random from a finite or infinite population.

**REFERENCES**

Dixon, W.J. (Ed.), *BMD: Biomedical Computer Programs*, (Univ.

Univ. of Notre Dame Computing Center, *BMD for the UNIVAC 1107*,
(Rough draft, 1967 ed. of BMD). Available from Univ. of
Notre Dame Computing Center, Notre Dame, Ind.

(John Wiley & Sons, Inc., New York, 1958). This text surveys
most of the procedures used by BMD Programs.

Cooley, W.W. and Lohnes, P.R., *Multivariate Procedures for the
Behavioral Sciences*, (John Wiley and Sons, Inc., New York,
1962). The MIT statistical package is described in this

continued
REFERENCES


SAMPLE INPUT AND OUTPUT

COST ESTIMATE

See references listed in USER INSTRUCTIONS.

Charge to user = computer time + postage and handling + network overhead

= computer time + $10.00 + network overhead

CONTENTS—BMD

pages
1 Identification & Abstract
3–12 User Instructions
13 I/O
15 Cost—Contents

6/70 15
UNIVAC 1107 Linear Programming Package

LP1107

The University of Notre Dame Computing Center

UNIVAC Division of Sperry Rand

Sleuth II/FORTRAN IV

UNIVAC 1107

Decks and listings presently available

Elizabeth Hutcheson, EIN Technical Representative, Computing Center, Univ. of Notre Dame, Notre Dame, Ind. 46556
Tel.: (219) 283-7784

LP1107 is a generalized program for the solution of linear programming problems. More specifically, the LP1107 command structure is a mathematical programming control language; used to prepare a control sequence for a specific programming job.

Either the dual or simplex algorithm, or both, can be used. LP1107 incorporates a true programming language with logical capabilities and full macro capabilities. If a group of commands is being used repetitively, the user may incorporate it into his macro command library and issue the macro command instead. Single and double precision arithmetic are included.

REFERENCES


Copies of these references are available through the local UNIVAC representatives or from the EIN Office at the cost of reproduction and mailing.
USER INSTRUCTIONS

The user is directed to the references for a full description of the operating system and programming language.

REFERENCES

UNIVAC Systems Programming Library Services, UP3897 1107LP


Copies of these references are available through the local UNIVAC representatives or from the EIN Office at the cost of reproduction and mailing.
SAMPLE INPUT

The problem shown is that described in *Introduction to Linear Programming* by Charnes, Cooper and Henderson, (John Wiley & Sons, New York, 1953).

```
RUN SMITH, 077707,5,20
XQT CUR
TRW F
IN F
TRW F
XQT 1107LP
SET CASE1 TO (NUT-MI) ; DEFINE CONTENTS OF CASE1
SET CASE2 TO (X ) ; DEFINE CONTENTS OF CASE2
TITLE ; NEXT CARD CONTAINS PAGE HEADING
1107LP RUN ON NUT-MIX PROBLEM
LOAD ; LOAD SHARE FORMAT MATRIX
ROW ID
1  +MA-CAS
1  +MA+PEA
1  +MB-CAS
1  +MB+PEA
1  +XX+CAS
1  +XX+PEA
1  +XX+HAZ
MATRIX
   XX+CAS   1.0
   XX+PEA   1.0
   XX+HAZ   0.6
   MA/CAS### $  0.5
   MA/CASMA-CAS -0.5
   MA/CASXX+CAS  1.0
   MA/PEA### $  0.5
   MA/PEAMA-CAS  0.5
   MA/PEAMA+PEA  0.75
   MA/PEAXX+PEA  1.0
   MA/HAZ### $  0.5
   MA/HAZMA-CAS  0.5
   MA/HAZMA+PEA -0.25
   MA/HAZXX+HAZ  1.0
   MB/CAS### $  0.35
   MB/CASMB-CAS -0.75
   MB/CASMB+PEA -0.5
   MB/CASXX+CAS  1.0
```

continued
MB/PEA$$$$$$ 0.35
MB/PEAMB-CAS 0.25
MB/PEAMB+PEA 0.5
MB/PEAXX+PEA 1.0
MB/HAZ$$$$$$ 0.35
MB/HAZMB-CAS 0.25
MB/HAZMB+PEA -0.50
MB/HAZXX+HAZ 1.0
MD/CAS$$$$$$ 0.25
MD/CAXX+CAS 1.0
MD/PEA$$$$$$ 0.25
MD/PEAXX+PEA 1.0
MD/HAZ$$$$$$ 0.25
MD/HAZXX+HAZ 1.0

**SLACK PRICES**
XX+CAS$$$$$$ 0.65
XX+PEA$$$$$$ 0.25
XX+HAZ$$$$$$ 0.35

ENDATA
LISTPR ; OUTPUT LABELS, COSTS, AND STATUS FLAGS
MATRIX ; OUTPUT MATRIX TABLEAU WITHOUT UNIT POSITIVES.
EQLIST ; OUTPUT MATRIX IN EQUATION FORM.
MIXMAP ; OUTPUT MATRIX IN COMPACT, CODED FORM.
GOGOGO ; OBTAIN OPTIMAL, FEASIBLE; RATIONAL SOLUTION.
DPMODE ; INVERT TO CURRENT BASIS IN DOUBLE PRECISION.
SPMODE ; RETURN TO SINGLE PRECISION.
GOGOGO ; RESOLVE IN CASE INVERSION INCOMPLETE.
PRIMAL ; OUTPUT ACTIVITY LEVELS AT SOLUTION.
DUAL ; OUTPUT SHADOW PRICES.
REDCST ; OUTPUT REDUCED COSTS OF NON BASIS VARIABLES.
RANGES ; OUTPUT PRIMAL AND DUAL BASIS.
ERRORS ; COMPUTE AND OUTPUT PRIMAL AND DUAL ERRORS.
ENDJOB ; RETURN TO MONITOR.
\)
SAMPLE OUTPUT

XQT 1107LP

SET CASE1 TO (NUT-MIX)  DEFINE CONTENTS OF CASE1
SET CASE2 TO (X )  DEFINE CONTENTS OF CASE2
TITLE  NEXT CARD CONTAINS PAGE HEADING
LOAD  LOAD SHARE FORMAT MATRIX

1107LP RUN ON NUT-MIX PROBLEM

LOADER STATISTICS
STRUCTURAL MATRIX
  7 ROWS 9 COLUMNS 21 NON-ZERO ELEMENTS 33.3 PCT. DENSE
  .250000 IS SMALLEST ELEMENT  1.000000 IS LARGEST ELEMENT

COST VECTOR
  12 NON-ZERO ELEMENTS  75.0 PCT. DENSE
  .250000 IS SMALLEST ELEMENT  .650000 IS LARGEST ELEMENT

RHS VECTOR
  3 NON-ZERO ELEMENTS  42.9 PCT. DENSE
  .600000 IS SMALLEST ELEMENT  1.000000 IS LARGEST ELEMENT
  0 INITIAL INFEASIBILITIES  0 INITIAL ARTIFICIALS

1107LP RUN ON NUT-MIX PROBLEM

LIST PRINT REPORT

FLAG FUNCTION
  1-ORIGINAL BASIS
  2-CURRENT BASIS
  3-DESIRED BASIS
  4-FREE VARIABLE
  5-FROZEN VARIABLE
  6-INVERSION CONTROL

continued
PVNO IS THE PIVOT VECTOR COUNT AT THE TIME THE VARIABLE LEFT THE BASIS

<table>
<thead>
<tr>
<th>LABEL</th>
<th>COST</th>
<th>FLAG</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>PVNO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 MA-CAS</td>
<td>.000000</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>2 MA+PEA</td>
<td>.000000</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>3 MB-CAS</td>
<td>.000000</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>4 MB+PEA</td>
<td>.000000</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>5 XX+CAS</td>
<td>.650000</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>6 XX+PEA</td>
<td>.250000</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>7 XX+HAZ</td>
<td>.350000</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>8 MA/CAS</td>
<td>.500000</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>9 MA/PEA</td>
<td>.500000</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>10 MA/HAZ</td>
<td>.500000</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>11 MB/CAS</td>
<td>.350000</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>12 MB/PEA</td>
<td>.350000</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>13 MB/HAZ</td>
<td>.350000</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>14 MD/CAS</td>
<td>.250000</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>15 MD/PEA</td>
<td>.250000</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
<tr>
<td>16 MD/HAZ</td>
<td>.250000</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>0</td>
</tr>
</tbody>
</table>

END OF LIST PRINT.

1107LP RUN ON NUT-MIX PROBLEM

CASE NUT-MIX

<table>
<thead>
<tr>
<th>LABEL</th>
<th>COST</th>
<th>BASIS</th>
<th>MA/CAS</th>
<th>MA/PEA</th>
<th>MA/HAZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 MA-CAS</td>
<td>.000000</td>
<td>.000000</td>
<td>-.50000</td>
<td>.500000</td>
<td>.500000</td>
</tr>
<tr>
<td>2 MA+PEA</td>
<td>.000000</td>
<td>.000000</td>
<td>-.25000</td>
<td>.750000</td>
<td>-.250000</td>
</tr>
<tr>
<td>3 MB-CAS</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
</tr>
<tr>
<td>4 MB+PEA</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
</tr>
<tr>
<td>5 XX+CAS</td>
<td>.650000</td>
<td>1.000000</td>
<td>1.000000</td>
<td>1.000000</td>
<td>1.000000</td>
</tr>
<tr>
<td>6 XX+PEA</td>
<td>.250000</td>
<td>1.000000</td>
<td>.000000</td>
<td>1.000000</td>
<td>.000000</td>
</tr>
<tr>
<td>7 XX+HAZ</td>
<td>.350000</td>
<td>.600000</td>
<td>.000000</td>
<td>.000000</td>
<td>1.000000</td>
</tr>
</tbody>
</table>

continued
### 1107LP Run on Nut-Mix Problem

#### Case Nut-Mix

<table>
<thead>
<tr>
<th>LABEL</th>
<th>MB/CAS</th>
<th>MB/PEA</th>
<th>MB/HAZ</th>
<th>MD/CAS</th>
<th>MD/PEA</th>
</tr>
</thead>
<tbody>
<tr>
<td>COST</td>
<td>.350000</td>
<td>.350000</td>
<td>.350000</td>
<td>.250000</td>
<td>.250000</td>
</tr>
<tr>
<td>Row Label</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 MA-CAS</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
</tr>
<tr>
<td>2 MA+PEA</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>.000000</td>
</tr>
<tr>
<td>3 MB-CAS</td>
<td>-.750000</td>
<td>.250000</td>
<td>.250000</td>
<td>.000000</td>
<td>.000000</td>
</tr>
<tr>
<td>4 MB+PEA</td>
<td>-.500000</td>
<td>.500000</td>
<td>-.500000</td>
<td>.000000</td>
<td>.000000</td>
</tr>
<tr>
<td>5 XX+CAS</td>
<td>1.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>1.000000</td>
<td>.000000</td>
</tr>
<tr>
<td>6 XX+PEA</td>
<td>.000000</td>
<td>1.000000</td>
<td>.000000</td>
<td>.000000</td>
<td>1.000000</td>
</tr>
<tr>
<td>7 XX+HAZ</td>
<td>.000000</td>
<td>.000000</td>
<td>1.000000</td>
<td>.000000</td>
<td>.000000</td>
</tr>
</tbody>
</table>

#### Case Nut-Mix (continued)

<table>
<thead>
<tr>
<th>LABEL</th>
<th>MD/HAZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>COST</td>
<td>.250000</td>
</tr>
<tr>
<td>Row Label</td>
<td></td>
</tr>
<tr>
<td>1 MA-CAS</td>
<td>.000000</td>
</tr>
<tr>
<td>2 MA+PEA</td>
<td>.000000</td>
</tr>
<tr>
<td>3 MB-CAS</td>
<td>.000000</td>
</tr>
<tr>
<td>4 MB+PEA</td>
<td>.000000</td>
</tr>
<tr>
<td>5 XX+CAS</td>
<td>.000000</td>
</tr>
<tr>
<td>6 XX+PEA</td>
<td>.000000</td>
</tr>
<tr>
<td>7 XX+HAZ</td>
<td>1.000000</td>
</tr>
</tbody>
</table>

**END of Matrix Tableau**
1107LP RUN ON NUT-MIX PROBLEM

EQLIST OUTPUT MATRIX IN EQUATION FORM.

1107LP RUN ON NUT-MIX PROBLEM

CASE NUT-MIX EQLIST OUTPUT

EQUATION 1 LABEL MA-CAS COST = .000000 ORIGINAL BI = .000000
+ 1.00000(MA-CAS) = .50000(MA/CAS) + .50000(MA/PEA) + .50000(MA/HAZ)

EQUATION 2 LABEL MA+PEA COST = .000000 ORIGINAL BI = .000000
+ 1.00000(MA+PEA) = .25000(MA/CAS) + .75000(MA/PEA) = .25000(MA/HAZ)

EQUATION 3 LABEL MB-CAS COST = .000000 ORIGINAL BI = .000000
+ 1.00000(MB-CAS) = .75000(MB/CAS) + .25000(MB/PEA) + .25000(MB/HAZ)

EQUATION 4 LABEL MB+PEA COST = .000000 ORIGINAL BI = .000000
+ 1.00000(MB+PEA) = .50000(MB/CAS) + .50000(MB/PEA) + .50000(MB/HAZ)

EQUATION 5 LABEL XX+CAS COST = .650000 ORIGINAL BI = 1.000000
+ 1.00000(XX+CAS) + 1.00000(MA/CAS) + 1.00000(MB/CAS) + 1.00000(MD/CAS)

EQUATION 6 LABEL XX+PEA COST = .250000 ORIGINAL BI = 1.000000
+ 1.00000(XX+PEA) + 1.00000(MA/PEA) + 1.00000(MB/PEA) + 1.00000(MD/PEA)

EQUATION 7 LABEL XX+HAZ COST = .350000 ORIGINAL BI = .600000
+ 1.00000(XX+HAZ) + 1.00000(MA/HAZ) + 1.00000(MB/HAZ) + 1.00000(MD/HAZ)

END EQLIST OUTPUT

continued
1107LP RUN ON NUT-MIX PROBLEM

MTXMAP OUTPUT

CASE NUT-MIX  ITERATION 0 OBJECTIVE VALUE .000000

C B M M M M M M M M M
O A A A A B B B D D D
S S / / / / / / / / /
T I C P H C P H C P H
S A E A A E A A A E A
S A Z S A Z S A Z

COST . . 5 5 5 5 5 5 5 5
MA-CAS .-5 5 5 . . . .
MA+PEA .-5 5-5 . . . .
MB-CAS . . -5 5 5 . .
MB+PEA . . . -5 5-5 . .
XX+CAS 5 1 1 . 1 . 1
XX+PEA 5 1 . 1 . 1 . 1
XX+HAZ 5 5 . 1 . 1 . 1

1107LP RUN ON NUT-MIX PROBLEM

THE MATRIX ELEMENTS ARE REPRESENTED AS FOLLOWS

CODE MAGNITUDE OF ELEMENTS OCCURRENCES

GREATER EQUAL TO OR THAN LESS THAN

0 0.0 0.0001 0
2 0.0001 0.001 0
3 0.001 0.01 0
4 0.01 0.1 0
5 0.1 0.9999 25
6 =1.0 11
7 1.0 10.0 0
8 10.0 100.0 0
9 100.0 1000.0 0

END MTXMAP OUTPUT

continued
GOOGOGO

OBTAIN OPTIMAL, FEASIBLE, RATIONAL SOLUTION.

STAGE 0
PVCNT 0
OBFVAL 1.110000
NET CHANGES 0

ITCNT 0
DJBAD 1000.000
KOLAY 9
NOLAY 9
ART 0
INF 0

96
MS FOR DOZ

135
MS FOR SSS

26
MS FOR UPDATE

65
MS FOR OPTMIZ

78
MS FOR PIVGEN

STAGE 1
PVCNT 5
OBFVAL 1.160000
NET CHANGES 5

ITCNT 5
DJBAD 1000.000
KOLAY 0
NOLAY 0
ART 0
INF 0

220
MS FOR DOZ

111
MS FOR SSS

49
MS FOR UPDATE

32
MS FOR OPTMIZ

4
MS FOR PIVGEN

STAGE 2
PVCNT 5
OBFVAL 1.160000
NET CHANGES 5

DPMODE
INVERSION
SPMODE
GOOGOGO

INVERT TO CURRENT BASIS IN DOUBLE PRECISION.
FINISHED WITH 0 NOT REMOVED AND 0 NOT ENTERED
RETURN TO SINGLE PRECISION
RESOLVE IN CASE INVERSION INCOMPLETE.

STAGE 0
PVCNT 5
OBFVAL 1.160000
NET CHANGES 5

ITCNT 5
DJBAD 1000.000
KOLAY 9
NOLAY 9
ART 0
INF 0

99
MS FOR DOZ

137
MS FOR SSS

94
MS FOR UPDATE

33
MS FOR OPTMIZ

2
MS FOR PIVGEN

STAGE 1
PVCNT 5
OBFVAL 1.160000
NET CHANGES 5

PRIMAL
OUTPUT ACTIVITY LEVELS AT SOLUTION.

continued
### PRIMAL OUTPUT

**CASE NUT-MIX**

<table>
<thead>
<tr>
<th>LABEL</th>
<th>COST</th>
<th>ACTIVITY</th>
<th>LABEL</th>
<th>COST</th>
<th>ACTIVITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>XX+PEA</td>
<td>.250000</td>
<td>.500000</td>
<td>XX+HAZ</td>
<td>.350000</td>
<td>.100000</td>
</tr>
<tr>
<td>MA/CAS</td>
<td>.500000</td>
<td>1.000000</td>
<td>MA/PEA</td>
<td>.500000</td>
<td>.500000</td>
</tr>
<tr>
<td>MA/HAZ</td>
<td>.500000</td>
<td>.500000</td>
<td>MB/PEA</td>
<td>.350000</td>
<td>.000000</td>
</tr>
<tr>
<td>MB/HAZ</td>
<td>.350000</td>
<td>.000000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**COST**

1.160000

**ACTIVITY**

### DUAL OUTPUT

**CASE NUT-MIX**

<table>
<thead>
<tr>
<th>LABEL</th>
<th>COST</th>
<th>SHADOW PRICE</th>
<th>LABEL</th>
<th>COST</th>
<th>SHADOW PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA-CAS</td>
<td>.000000</td>
<td>.350000</td>
<td>MA+PEA</td>
<td>.000000</td>
<td>.100000</td>
</tr>
<tr>
<td>MB-CAS</td>
<td>.000000</td>
<td>.200000</td>
<td>MB+PEA</td>
<td>.000000</td>
<td>.100000</td>
</tr>
<tr>
<td>XX+CAS</td>
<td>.650000</td>
<td>.700000</td>
<td>XX+PEA</td>
<td>.250000</td>
<td>.250000</td>
</tr>
<tr>
<td>XX+HAZ</td>
<td>.350000</td>
<td>.350000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**COST**

1.160000

**SHADOW PRICE**

### REDCST OUTPUT

**CASE NUT-MIX**

<table>
<thead>
<tr>
<th>LABEL</th>
<th>COST</th>
<th>REDUCED COST</th>
<th>LABEL</th>
<th>COST</th>
<th>REDUCED COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA-CAS</td>
<td>.000000</td>
<td>.350000</td>
<td>MA+PEA</td>
<td>.000000</td>
<td>.100000</td>
</tr>
<tr>
<td>MB-CAS</td>
<td>.000000</td>
<td>.200000</td>
<td>MB+PEA</td>
<td>.000000</td>
<td>.100000</td>
</tr>
<tr>
<td>XX+CAS</td>
<td>.650000</td>
<td>.050000</td>
<td>MB/CAS</td>
<td>.350000</td>
<td>.150000</td>
</tr>
<tr>
<td>MD/CAS</td>
<td>.250000</td>
<td>.450000</td>
<td>MD/PEA</td>
<td>.250000</td>
<td>.000000</td>
</tr>
<tr>
<td>MD/HAZ</td>
<td>.250000</td>
<td>.100000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**COST**

1.160000

**REDUCED COST**

*continued*
### Dual Range Output

**CASE NUT-MIX** | **ITERATION** | 5 | **OBJECTIVE VALUE** | 1.160000
--- | --- | --- | --- | ---

<table>
<thead>
<tr>
<th>LABEL</th>
<th>ORIG. ACT.</th>
<th>LABEL</th>
<th>INCREMENT</th>
<th>LABEL</th>
<th>INCREMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA-CAS</td>
<td>.000000</td>
<td>XX+HAZ</td>
<td>-.066667</td>
<td>MA/HAZ</td>
<td>.333333</td>
</tr>
<tr>
<td>MA+PEA</td>
<td>.000000</td>
<td>MA/HAZ</td>
<td>-.500000</td>
<td>XX+HAZ</td>
<td>.100000</td>
</tr>
<tr>
<td>MB-CAS</td>
<td>.000000</td>
<td>XX+HAZ</td>
<td>-.050000</td>
<td>MB/HAZ</td>
<td>.000000</td>
</tr>
<tr>
<td>MB+PEA</td>
<td>.000000</td>
<td>MB/HAZ</td>
<td>-.000000</td>
<td>MB/PEA</td>
<td>.000000</td>
</tr>
<tr>
<td>XX+CAS</td>
<td>1.000000</td>
<td>XX+HAZ</td>
<td>-.200000</td>
<td>MA/HAZ</td>
<td>1.000000</td>
</tr>
<tr>
<td>MB/CAS</td>
<td>.000000</td>
<td>MB/HAZ</td>
<td>-.000000</td>
<td>XX+HAZ</td>
<td>.100000</td>
</tr>
<tr>
<td>MD/CAS</td>
<td>.000000</td>
<td>XX+HAZ</td>
<td>-.200000</td>
<td>MA/HAZ</td>
<td>.100000</td>
</tr>
<tr>
<td>MD/PEA</td>
<td>.000000</td>
<td>MA/HAZ</td>
<td>.000000</td>
<td>XX+PEA</td>
<td>.500000</td>
</tr>
<tr>
<td>MD/HAZ</td>
<td>.000000</td>
<td>XX+PEA</td>
<td>-.9999.000000</td>
<td>XX+HAZ</td>
<td>.100000</td>
</tr>
</tbody>
</table>

**End Dual Range Output**

### Primal Range Output

**CASE NUT-MIX** | **ITERATION** | 5 | **OBJECTIVE VALUE** | 1.160000
--- | --- | --- | --- | ---

<table>
<thead>
<tr>
<th>LABEL</th>
<th>COST</th>
<th>LABEL</th>
<th>INCREMENT</th>
<th>LABEL</th>
<th>INCREMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>XX+PEA</td>
<td>.250000</td>
<td>MD/PEA</td>
<td>-.000000</td>
<td>MA+PEA</td>
<td>.100000</td>
</tr>
<tr>
<td>XX+HAZ</td>
<td>.350000</td>
<td>MA+PEA</td>
<td>-.100000</td>
<td>MB-CAS</td>
<td>.100000</td>
</tr>
<tr>
<td>MA/CAS</td>
<td>.500000</td>
<td>XX+CAS</td>
<td>-.050000</td>
<td>9999.000000</td>
<td>9999.000000</td>
</tr>
<tr>
<td>MA/PEA</td>
<td>.500000</td>
<td>MA+PEA</td>
<td>-.100000</td>
<td>9999.000000</td>
<td>9999.000000</td>
</tr>
<tr>
<td>MA/HAZ</td>
<td>.500000</td>
<td>XX+CAS</td>
<td>-.100000</td>
<td>MA+PEA</td>
<td>.100000</td>
</tr>
<tr>
<td>MB/PEA</td>
<td>.350000</td>
<td>MB-CAS</td>
<td>-.100000</td>
<td>MB/CAS</td>
<td>.075000</td>
</tr>
<tr>
<td>MB/HAZ</td>
<td>.350000</td>
<td>MB-CAS</td>
<td>-.100000</td>
<td>MB+PEA</td>
<td>.100000</td>
</tr>
</tbody>
</table>

**End Primal Range Output**

*continued*
### Dual Error Analysis

<table>
<thead>
<tr>
<th>LABEL</th>
<th>Z(J)</th>
<th>C(J)</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>XX+PEA</td>
<td>.250</td>
<td>.250</td>
<td>.000000000</td>
</tr>
<tr>
<td>XX+HAZ</td>
<td>.350</td>
<td>.350</td>
<td>.000000000</td>
</tr>
<tr>
<td>MA/CAS</td>
<td>.500</td>
<td>.500</td>
<td>.000000000</td>
</tr>
<tr>
<td>MA/HAZ</td>
<td>.500</td>
<td>.500</td>
<td>.000000000</td>
</tr>
<tr>
<td>MB/PEA</td>
<td>.350</td>
<td>.350</td>
<td>.000000000</td>
</tr>
<tr>
<td>MB/HAZ</td>
<td>.350</td>
<td>.350</td>
<td>.000000000</td>
</tr>
</tbody>
</table>

**Maximum Dual Error of .00000000 For Variable 1107LP Run on Nut-Mix Problem**

### Primal Error Analysis

<table>
<thead>
<tr>
<th>LABEL</th>
<th>ORIG'B(I)</th>
<th>CALC B(I)</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA-CAS</td>
<td>.000</td>
<td>.000</td>
<td>-.000000000</td>
</tr>
<tr>
<td>MA+PEA</td>
<td>.000</td>
<td>.000</td>
<td>-.000000000</td>
</tr>
<tr>
<td>MB-CAS</td>
<td>.000</td>
<td>.000</td>
<td>-.000000000</td>
</tr>
<tr>
<td>MB+PEA</td>
<td>.000</td>
<td>.000</td>
<td>-.000000000</td>
</tr>
<tr>
<td>XX+CAS</td>
<td>1.000</td>
<td>1.000</td>
<td>-.000000000</td>
</tr>
<tr>
<td>XX+PEA</td>
<td>1.000</td>
<td>1.000</td>
<td>-.000000000</td>
</tr>
<tr>
<td>XX+HAZ</td>
<td>.600</td>
<td>.600</td>
<td>-.000000000</td>
</tr>
</tbody>
</table>

**Maximum Primal Error of .00000000 For Variable 1107LP Run on Nut-Mix Problem**

ENDJOB  
RETURN TO MONITOR.

**Run Time:** 0 HRS 0 MIN 21 SEC
COST ESTIMATE

For the program listed on the Sample Input, computer time was 21 seconds. At the current rate for the University of Notre Dame ($480./hr. plus input-output charges), the chargeable computer cost was $4.18.

Charge to user = computer costs + postage and handling + network overhead

= $4.18 + $10.00 + network overhead

= $14.18 + network overhead

CONTENTS—LP1107

<table>
<thead>
<tr>
<th>pages</th>
<th>Identification &amp; Abstract</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>User Instructions</td>
<td></td>
</tr>
<tr>
<td>5-15</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Cost—Contents</td>
<td></td>
</tr>
</tbody>
</table>
UNIVAC 1107 PERT/COST, PERT/TIME System

PERT (library tape)

The University of Notre Dame Computing Center

UNIVAC Division of Sperry Rand Corporation

SLEUTH II and COBOL

UNIVAC 1107

Listings available

Elizabeth Hutcheson, EIN Technical Representative, Computing Center, University of Notre Dame, Notre Dame, Ind. 46556
Tel. (219) 283-7784

The UNIVAC 1107 PERT System provides for integrated time and cost planning and control of research and development programs through implementation of the "work package" costing concept of the Department of Defense/National Aeronautics and Space Administration. The system is composed of two major program modules, PERT/COST and PERT/TIME, which generate reports for the project being analyzed. These reports include activity and event reports, a Work package/Activity listing, a Charge number analysis, etc., from PERT/TIME, and reports on the structure of labor and material costs from PERT/COST.

REFERENCES


Available through the local UNIVAC representative or through the EIN Office at the cost of reproduction and mailing.
USER INSTRUCTIONS
Potential users are directed to the references listed below for a full discussion of the options and constraints in effect.

REFERENCES


Available through the local UNIVAC representative or through the EIN Office at the cost of reproduction and mailing.
SAMPLE INPUT and SAMPLE OUTPUT
Sample I/O is included in the references listed in the USER INSTRUCTIONS; interested persons are directed to these documents.
COST ESTIMATE
See references listed in USER INSTRUCTIONS.

CONTENTS—PERT
pages
1 Identification & Abstract
3 User Instructions
5 I/O
7 Cost—Contents
FUNCTIONAL ABSTRACT

The Inquirer II is a set of computer programs comparable to, but more flexible than, its predecessor, the General Inquirer, developed by Stone and his colleagues at Harvard. The original version of the General Inquirer System was designed for problems encountered in the content analysis of textual and verbal data. The General Inquirer was implemented for the IBM 7090-7094 computer along with the IBM 1401 computer. A later version of the General Inquirer was designed by Psathas and Miller to be used only with an IBM 1401 computer with an IBM 1311 disk drive. Stone originally described the General Inquirer as "a set of computer programs to (a) identify, systematically within text, instances of words and phrases that belong to categories specified by the investigator; (b) count occurrences and specify co-occurrences of these categories; (c) print and graph tabulations; (d) perform statistical tests; (e) sort and regroup sentences according to whether they contain instances of a particular category of combination of categories." The Inquirer II contains these capabilities and also
allows for more elaborate analysis of the data. The I/II is able to make more elaborate contextual searches of the data and provide more options to the potential users. These options are described in detail in the Inquirer II Programmer's Guide.

Content analysis may be defined as a research technique which includes a systematic identification of theoretically relevant constructs in textual data. Content analysis is usually performed so that inferences can be made about the source or originator of the message, the message itself or the intended receiver of the message. The investigator communicates the constructs and the rules by which they may be identified within the corpus of text by means of dictionary. This dictionary is either one of his own construction or one which has been utilized in previous research.

REFERENCES


USER INSTRUCTIONS

The primary inputs to the system are the data to be analyzed and a dictionary of concepts.

The major task in using the Inquirer II system is the creation of a dictionary of content analysis categories, called concepts. A concept consists of a number of language signs that together represent a variable in the investigator's theory.

The Sample Output appended gives a general idea of the process of presenting the data and formulating the dictionary of concepts. A thorough understanding for the user will be obtained by sending for the complete documentation, in particular the Inquirer II Programmer's Guide.¹

REFERENCES

SAMPLE INPUT

PROTOCOL * NEW TITLE ENCOUNTERED

Silene's delusional system from Autobiography of a Schizophrenic Girl, in Landis p 182-38

Very soon after the beginning of analysis I understood that my fear was a cover for guilt, a guilt infinite and awful. During the early sessions, masturbation and the hostility I harbored toward everyone seemed to lie at the bottom. I literally hated people, without knowing why. In dreams and frequently in waking fantasies I constructed an electric machine to blow up the earth and everyone with it. This was my greatest, most terrible revenge.

Later, considering them appropriate, I no longer felt guilty about these fantasies, nor did the guilt have an actual object. It was too pervasive, too anxious, to be founded on anything definite, and it demanded punishment. The punishment was indeed horrible, sadistic — it consisted, fittingly enough, of being guilty. For to feel oneself guilty is the worst that can happen, it is the punishment of punishments. Consequently, I could never be relieved of it as though I had been truly punished. Quite the reverse, I felt more and more guilty, inexpressibly guilty. Consequently, I sought to discover what was punishing me so dreadfully, what was making me so guilty.
other beings. A formidable interdependence bound all men under the scourge of culpability. Everyone was guilty if it was some vast world-like entity encompassing all men. At the top were those who gave orders, who were guilty. Since every man was responsible for all other men, each of his acts had a repercussion.

A desk like a puppet, separated from everything, alone under a blinding light, waving his arms like a maniac. And I saw your little sister, rolling on the kitchen floor like a puppet, separated from everything, alone under a blinding light, waving his arms like a maniac.

Very soon after the beginning of analysis I understood that my fear was a cover for guilt, a trait infinite and awful. During the early sessions, masturbation was the greatest, most unbearable, to be found on anything definite, and it demanded punishment. The punishment was indestructible.

During the early sessions, masturbation and the hostility I harbored toward everyone seemed to me, abominable, intolerably guilty, without cause and without active. Any punishment, the very worst, could be imposed on me.
Dictionary Compiler output--

Concept Name Paragraph

BEGIN COMPILATION OF SOURCE PROGRAM

AGE=01,
SEX=02,
SOMATIC=03,
ILLNESS=04,
DEATH=05,
TEST=06,
NON_SPEC=07,
TEXTUAL=08,
LEGAL=09,
STATEMENT=10,
ACTION=11,
FACILITY=12,
ATTRIBUTES=13,
REFERENT=14,
RELATION=15,
POLICY=16,
PATTERN=17,
SCHEDULE=18,
SPECIAL=19,
SEX=20,
SEXUAL=21,
SEXUAL_ACTIVITY=22,
PREFERENCE=23,
RELATIONSHIP=24,
SPECIALIZATION=25,
PERCEPTION=26,
HORMONAL=27,
FACULTY=28,
SCHOOL=29,
PERSON=30,
PERSONAL=31,
ROLE=32,
FUNCTION=33,
CHARACTERISTIC=34,
STATEMENT=35,
AFFIRMATION=36,
AFFIRMATIVE=37,
AFFIRMATIVE=38,
AFFIRMATIVE=39,
AFFIRMATIVE=40,
AFFIRMATIVE=41,
AFFIRMATIVE=42,
AFFIRMATIVE=43,
AFFIRMATIVE=44,
AFFIRMATIVE=45,
AFFIRMATIVE=46,
AFFIRMATIVE=47,
AFFIRMATIVE=48,
AFFIRMATIVE=49,
AFFIRMATIVE=50,
AFFIRMATIVE=51,
AFFIRMATIVE=52,
AFFIRMATIVE=53,
AFFIRMATIVE=54,
AFFIRMATIVE=55,
AFFIRMATIVE=56,
AFFIRMATIVE=57,
AFFIRMATIVE=58,
AFFIRMATIVE=59,
AFFIRMATIVE=60,
AFFIRMATIVE=61,
AFFIRMATIVE=62.

continued
Dictionary Compiler-- List of Concept

Name Paragraph and start of Entry Name Paragraph

* AUGER.63,64.
* LOVE=63,65.
* HATE=63,66.
* FEAR=63,67.
* HAPPY=63,68.
* SAD=63,69.
* DISTRESS=63,70.
* EMOTION_POS63,71.
* EMOTION_NEG63,72.
* INDEPENDENT_NEED73.
* OBSTACLES-74.
* OBST_PRES=74,78.
* STRUGGLE-74,76.
* SELF=77.
* SPEC_OTHERS78.
* NON_SPEC_OTHERS=79.
* SENT_LENGTH 90.
* LONG_SENT=90,91.
* SHORT_SENT=90,92.
* SENT_TYPE=93.
* QUESTION=93,94.

ABOUT: IF WORD (-4,-2) = 'UP' & WORD (-3,-1) = 'AND'
THEN HEALTH
ELSE IF WORD (-2,-1) = 'BROUGHT'
THEN SUCCESS
ELSE IF WORD (1,2) = 'TO'
THEN ACTIVE
ELSE.

ACT: IF WORD (-2,-1) = 'SEXUAL'
THEN AGGRESSIVE
ELSE IF WORD (1,1) = 'UPON'
THEN AGGRESSIVE
ELSE.

ALL: IF ORDER(WORD (-3,-2) = 'FREE', WORD (-2,-1) = 'FOR')
THEN AGGRESSIVE
ELSE IF WORD (1,3) = 'OVER'
THEN ACHIEVE
ELSE.

ALONE: ISOLATE.

ANALYSIS: CONTEMPLATE.

ANALYST: TREAT_FIG: NEUTER.

ANGUISH: SAD: EMOTION_NEG.

ANY: NON_SPEC_REF.

ANYTHING: NON_SPEC_REF.

APPEARANT: DECIDE.

APPARENTLY:

APPROPRIATE:

ARM:

ARMS:

ASSOCIATE: IF WORD (-2,-1) = 'AN'
THEN AGGRESSIVE
ELSE IF ORDER(WORD (2,-1) = 'THE', WORD (-2,-1) = 'THE',
THEN AGGRESSIVE
ELSE.

AT: IF WORD (-3,-1) = 'FLY'
THEN AGGRESSIVE

continued
Portion of Entry Name Paragraph of Dictionary

```plaintext
LOL IF WORD (1,4) = 'HELL'
THEN HEALTH
ELSE.

FILT:
IF WORD (1,2) = 'FOR'
THEN (LOL: PORTION_POS)
ELSE IF WORD (1,2) = 'WELL'
THEN HEALTH
ELSE.

FIGURE:
IF WORD (9,11) = 'I' | WORD (10,11) = 'THE'
AND WORD (11,11) = 'BEH' |...
THEN HEALTH
ELSE.

FIT:
IF WORD (1,2) = 'THE'
AND WORD (2,3) = 'BEH'
THEN HEALTH
ELSE.

FOUND:
IF WORD (1,4) = 'OUT'
THEN SUCCESS
ELSE.

FROM:
IF WORD (-3,-2) = 'SHAPES' | WORD (-3,-1) = 'SHRINK'
AND ORDER(WORD (-3,-1) = 'OUT'
AND WORD (-4,-1) = 'BEH'
AND WORD (1,3) = 'BEH'
THEN HEALTH
ELSE IF WORD (-2,-2) = 'GET' | WORD (-1,-1) = 'BACK'
THEN APPROACH
ELSE.

GATE:
IF WORD (1,2) = 'IN' | WORD (1,3) = 'OUT'
AND WORD (1,5) = 'WAY'
THEN SUBMIT
ELSE EXPELL.

GET:
IF WORD (1,1) = 'AROUND' | WORD (1,2) = 'UP'
THEN ACTIVE
ELSE IF WORD (1,2) = 'AWAY' | WORD (1,3) = 'OUT'
AND WORD (1,5) = 'OFF'
THEN AVOID
ELSE IF WORD (1,2) = 'BACK' | WORD (1,3) = 'IN'
AND WORD (1,5) = 'OUT'
AND WORD (1,5) = 'BEH'
THEN APPROACH
ELSE IF ORDER(WORD (1,2) = 'SHAPES', WORD (2,3) = 'OUT'
AND ORDER(WORD (1,3) = 'TO', WORD (2,4) = 'BOTTOM',
AND WORD (1,4) = 'THROUGH'
AND WORD (1,1) = 'AROUND', WORD (2,2) = 'TO'
THEN SUCCESS
ELSE IF WORD (1,3) = 'HOLD' | WORD (2,2) = 'UP'
AND ORDER(WORD (1,4) = 'IN', WORD (2,5) = 'WAY'
THEN SUCCESS
ELSE.

GET:
IF ORDER(WORD (1,2) = 'SHAPES', WORD (2,3) = 'OUT'
AND ORDER(WORD (1,3) = 'TO', WORD (2,4) = 'BOTTOM',
AND WORD (1,4) = 'THROUGH'
AND WORD (1,1) = 'AROUND', WORD (2,2) = 'TO'
THEN SUCCESS
ELSE IF WORD (1,3) = 'HOLD' | WORD (2,2) = 'UP'
AND ORDER(WORD (1,4) = 'IN', WORD (2,5) = 'WAY'
THEN SUCCESS
ELSE.

GET:
IF ORDER(WORD (1,2) = 'SHAPES', WORD (2,3) = 'OUT'
AND ORDER(WORD (1,3) = 'TO', WORD (2,4) = 'BOTTOM',
AND WORD (1,4) = 'THROUGH'
AND WORD (1,1) = 'AROUND', WORD (2,2) = 'TO'
THEN SUCCESS
ELSE IF WORD (1,3) = 'HOLD' | WORD (2,2) = 'UP'
AND ORDER(WORD (1,4) = 'IN', WORD (2,5) = 'WAY'
THEN SUCCESS
ELSE.
```

continued
CONCEPTS: Listing of Tagged File

Renee's delusional system from Autobiography of a Schizophrenic Girl

ID FIELD OF LEVEL 2 = p15
ID FIELD OF LEVEL 1 = p1

SENTENCE 1: 22 WORDS
CURRENT ID p1

Very soon after the beginning of analysis I understood that

SADNESS AS Acares for guilt, ANGER AS A Guilt, INFANTILE AND

AWFUL REGARD NEGATIVE

SENTENCE 2: 16 WORDS
CURRENT ID p1

Dancing the early waltz, masturbation and the hostility I self-hated

TOWARD EVERYONE ADRESSED AS I LIE AT THE BOTTOM AND

DIRECTION NON_SPEC_OTHRES APPROACH NEUTER COMMUNICATE

SENTENCE 3: 7 WORDS
CURRENT ID p1

I self literally hated people, knowing all a

SELF EMOTION NEGATIVE COG_PROC CONTINUE

SENTENCE 4: 21 WORDS
CURRENT ID p1

In dreams and frequently in making fantasies I constructed as

ELECTRIC RATCHETS TO BLOW SOCIAL_ACT ACTION NORM

SENTENCE 5: 7 WORDS
CURRENT ID p1

This was by self REGARD POSITIVE REGARD NEGATIVE

continued
Tabulation of tagged file

From "Delusional System from Autobiography of a Schizophrenic Girl," in Landis p 182-3

<table>
<thead>
<tr>
<th>CONCEPT NAME</th>
<th>TOTAL</th>
<th>TOTAL</th>
<th>CONCEPT WD - CONC</th>
<th>SENTENCE</th>
<th>SENTENCE</th>
<th>SENTENCE</th>
<th>SENTENCE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FREQ</td>
<td>INDEX</td>
<td>INDEX</td>
<td>INDEX</td>
<td>FREQ</td>
<td>INDEX</td>
<td>INDEX</td>
</tr>
<tr>
<td>AUDITORY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GUSTATORY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Olfactory</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dermal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Passive</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>SEX_ACT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FARK_PLAY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GEN_FORK_PLAT</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>AGGRESSIVE</td>
<td>2</td>
<td>2.66</td>
<td>3.50</td>
<td>6.89</td>
<td>2</td>
<td>40.00</td>
<td>40.00</td>
</tr>
<tr>
<td>FRIENDLY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISOLATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMUNICATE</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>DOMINATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUBMIT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HELP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>APPROACH</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>AVOID</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

continued
### Tabulation of tagged file

**Renee's delusional symptoms from Autobiography of a Schizophrenic Girl, in Landis p 182-3**

<table>
<thead>
<tr>
<th>CONCEPT_NAME</th>
<th>TOTAL_FREQ</th>
<th>WORD_INDEX</th>
<th>CONCEPT_INDEX</th>
<th>SB - CONC_INDEX</th>
<th>SENTENCE_FREQ</th>
<th>SENTENCE_INDEX</th>
<th>SNCC - CONC_INDEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUCCESS</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>FAIL</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>CONTEMPLATE</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>CDG_AWARE</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>UNCERTAIN</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>DECIDE</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>POSITIVE</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>NEGATIVE</td>
<td>2</td>
<td>2.66</td>
<td>3.50</td>
<td>6.89</td>
<td>2</td>
<td>40.00</td>
<td>40.00</td>
</tr>
<tr>
<td>GOOD_POST</td>
<td>2</td>
<td>2.66</td>
<td>3.50</td>
<td>6.89</td>
<td>2</td>
<td>40.00</td>
<td>40.00</td>
</tr>
<tr>
<td>BAD_POST</td>
<td>2</td>
<td>2.66</td>
<td>3.50</td>
<td>6.89</td>
<td>2</td>
<td>40.00</td>
<td>40.00</td>
</tr>
<tr>
<td>ANGER</td>
<td>4</td>
<td>2.66</td>
<td>3.50</td>
<td>6.89</td>
<td>2</td>
<td>40.00</td>
<td>40.00</td>
</tr>
<tr>
<td>LOVE</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>HATE</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>FEAR</td>
<td>3</td>
<td>4.00</td>
<td>5.26</td>
<td>10.34</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>HAPPY</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>SAD</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>DISTRESS</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
<tr>
<td>EMOTION_POS</td>
<td>1</td>
<td>1.33</td>
<td>1.75</td>
<td>3.44</td>
<td>1</td>
<td>20.00</td>
<td>20.00</td>
</tr>
</tbody>
</table>
COST ESTIMATE

It is difficult to offer a precise cost estimate for a given task. A full treatment of a 5000-word text would cost approximately $500.

CONTENTS—I/II

pages  
1–2 Identification & Abstract  
3 User Instructions  
5–12 I/O  
13 Cost—Contents
DESCRIPTIVE TITLE
Basic Information Retrieval System

CALLING NAME
BIRS

INSTALLATION NAME
Michigan State University
Information Systems Laboratory

AUTHOR(S) AND AFFILIATION(S)
John F. Vinsonhaler, Ph.D.
John M. Hafterson
Stuart W. Thomas, Jr.
Michigan State University

LANGUAGE
USASI Full FORTRAN

COMPUTER
CDC 3600, CDC 6500, CDC 6600
IBM 360 G-level
GE 600 Series, others

PROGRAM AVAILABILITY
Decks and listings are currently available at cost from Michigan State University for non-profit institutions.

The program is distributed to profit-making institutions by Hygain Technologies, 65 Whitney Street, Westport, Conn.

Maintenance for all users is provided by Hygain Technologies.

CONTACT
Dr. John F. Vinsonhaler, Director,
Information Systems Laboratory,
309 Computer Center, Michigan State University, East Lansing, Mich. 48823
Tel.: (517) 353-7284

FUNCTIONAL ABSTRACT
BIRS is a general purpose system of programs for the behavioral sciences and education. Essentially, BIRS is a set of fundamental program modules designed to allow scholars and scientists to use their own locally based computer to construct and maintain a variety of information systems. Search, maintenance, and index creation are performed automatically. Thus, BIRS may be viewed as a set of essential tools; the research worker may use these tools to construct the type of information system which best meets his immediate needs.

continued
It is not economically practical for individual educators and social scientists to develop their own special purpose programming system; they must share the costs of system developments by exchanging programs. General purpose systems like BIRS (with machine, data, and application independent programming) are ideally suited for free exchange among computer users.

REFERENCES


Documentation is available at cost from the Information Systems Laboratory and from Hygain Technologies.
USER INSTRUCTIONS

See the references cited below. The nature and complexity of the system can be studied in brief with the help of the samples of input and output following.

REFERENCES


Documentation is available at cost from the Information Systems Laboratory and from Hygain Technologies.
SAMPLE INPUT AND OUTPUT

Information Storage: Creating the Information File

INPUT FOR BIRS PROCESSING

\$IFMP
\$DELIMITER IS \#
\$NEW FILE
\$ABSTRACT
STATE MICHIGAN
PUBLIC LAW 89-10
TITLE AUGMENTATION OF SPECIAL EDUCATION AND PRE-EDVOCATIONAL TRAINING
PROGRAMS.

DESCRIPTORS EMOTIONALLY DISTURBED, READING DISABILITY, PREVOCATIONAL
EDUCATION, RESIDENTIAL SCHOOL
TOTAL CHILDREN 90
TOTAL EXPENDITURE \$47,000
INSTRUCTION SALARIES = \$39,658, MATERIALS = \$1,542, OTHER = 0
PROJECT PARTICIPANTS

NUMBER OF TEACHERS TMR EMR HH
5 3 0
OTHER PERSONNEL 0 0 2

SUMMARY
THIS PROJECT IS TO REDUCE THE NUMBER OF GRADE LEVELS PER CLASSROOM IN
OUR SCHOOL, PROVIDE GREATER INDIVIDUALIZATION OF INSTRUCTION WITHIN
THE CLASSROOMS AND, TO EXPOSE OUR EMOTIONALLY DISTURBED CHILDREN TO
GREATER SOCIALIZATION PROGRAMS

... other data elements preceded by \$ABSTRACT

\$END

OUTPUT FROM BIRS PROCESSING

INFORMATION FILE — CONTAINING DATA RECORDS

Printed
Filed
continued

6/70 5
Information Indexing: Preparing a Key Word Index

INPUT FOR BIRS PROCESSING

*$PIP
*$DELIMITER #
*$USE FIELDS

PUBLIC LAW,STATE
*$BOOK INDEX
*$END

OUTPUT FROM BIRS PROCESSING

Index for the Information File

* * * * * * * * * * * * * * * * * * * *

(PUBLIC LAW)

89-10 1, 22-60
89-313 2-21, 100-350
TITLE IV 50, 53, 500-650

(STATE)

ALABAMA 200-250, 301
ALASKA 10-11, 22

* * * * * * * * * * * * * * * * * * * *

continued
Information Retrieval: Preparing a List of Particular Types of Projects

INPUT FOR BIRS PROCESSING

$DFSP
$DELIMITER #
$QUESTION
  #STATE,MICHIGAN.AND. (#DESCRIPTORS, EMOTIONALLY DISTURBED OR.
  EMOTIONALLY HANDICAPPED)
$END
$IFRP
$DELIMITER #
$PRINT LINES 1 TO 5
$END

OUTPUT FROM BIRS PROCESSING

SEARCH REPORT

* * * * * * * * * * * * * * * * * * * * * * * * *

ABSTRACT 1  RELEVANCE 1.00
#STATE  MICHIGAN
#PUBLIC LAW  89-10
#TITLE  AUGMENTATION OF SPECIAL EDUCATION AND PRE-VOCATIONAL TRAINING
        PROGRAMS

ABSTRACT 35  RELEVANCE 1.00
#STATE  MICHIGAN
#PUBLIC LAW  TITLE IV-A
#TITLE  IMPROVEMENT OF SPECIAL ASSISTANCE FOR EMOTIONALLY DISTURBED

continued
Information Analysis: Preparing a Statistical Report

INPUT FOR BIRS PROCESSING

```
$IRPG
$DELIMITER #
$DIMENSION
STATES = #STATE,MICHIGAN,HAWAII,ILLINOIS,MISSOURI
$DIMENSION
PURPOSE = #INSTRUCTION, SALARIES, MATERIALS, TOTAL(SALARIES, MATERIALS)
$DEFINE DATA
COST = .SUM.#INSTRUCTION
$TABLE
ALLOCATION OF FUNDS FOR INSTRUCTION (COST) = PURPOSE, STATES
```

OUTPUT FROM BIRS PROCESSING

TABULAR REPORT

```
<table>
<thead>
<tr>
<th></th>
<th>MICHIGAN</th>
<th>HAWAII</th>
<th>ILLINOIS</th>
<th>MISSOURI</th>
</tr>
</thead>
<tbody>
<tr>
<td>SALARIES</td>
<td>80385.00</td>
<td>69717.00</td>
<td>71800.00</td>
<td>41154.00</td>
</tr>
<tr>
<td>MATERIALS</td>
<td>4642.00</td>
<td>826.00</td>
<td>5600.00</td>
<td>1011.00</td>
</tr>
<tr>
<td>TOTAL</td>
<td>85027.00</td>
<td>70543.00</td>
<td>77400.00</td>
<td>42165.00</td>
</tr>
</tbody>
</table>
```

---

PAGE 1

---

6/70
COST ESTIMATE

Unfortunately, cost estimates for the sample program are unavailable. The average cost for complete processing is about $15.00 per abstract per year for most applications, including file maintenance, searching, and report generation.

Charge to user = computer costs + network overhead

= $15.00/abstract per year + network overhead

CONTENTS—BIRS

page
1–2 Identification & Abstract
3 User Instructions
5–8 I/O
9 Cost—Contents
FUNCTIONAL ABSTRACT

It frequently happens that two factor solutions are obtained in a study, and the question arises as to the extent of similarity. Cliff's approach to the problem is succinctly stated in the abstract of his paper,

Two problems are considered. The first is that of rotating two factor solutions orthogonally to a position where corresponding factors are as similar as possible. A least-squares solution for transformations of the two factor matrices is developed. The second problem is that of rotating a factor matrix orthogonally to a specified target matrix.

The present program performs least squares, procrustes rotations on a target (T) and a data (D) matrix following the general approach of Cliff.

General Description

The mathematical basis for the program is summarized briefly in the following outline.
Case I: Find an A and B such that transformation
(E'E) = minimum
where
E = TA - DB.
A and B are given by
T'D = Q
Q = AΓB'
where A, B' and Γ are the Eckart and Young\(^2\) decomposition of Q.

Case II: Find a C such that transformation
(E'E) = minimum
where
E = T - DC
C is given by
C = B'A'

Case II is the typical target-matching situation were T is held fixed and D rotated to a least-squares fit.

The program can handle matrices as large as 125 rows (variables) X 50 columns.

REFERENCES
**USER INSTRUCTIONS**

**Card Preparation**

**Problem Card**

<table>
<thead>
<tr>
<th>Columns</th>
<th>Parameter</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–5</td>
<td>NR</td>
<td>I5</td>
<td>Number of rows (variables) ≤ 125</td>
</tr>
<tr>
<td>6–10</td>
<td>NCT</td>
<td>I5</td>
<td>Number of columns in target matrix (≤ 50)</td>
</tr>
<tr>
<td>11–15</td>
<td>NCD</td>
<td>I5</td>
<td>Number of columns in data matrix (≤ 50)</td>
</tr>
</tbody>
</table>
| 16–20   | ICASE     | I5     | Case option
1: Simultaneous rotations
2: Target matching
0: Both 1 and 2 |
| 21–25   | IFMT      | I5     | Format option
0: Use same format for target and data matrices
≠ 0: Read in a different format for the data matrix |

**Title Card**

Name of problem in Cols. 1–80

**Format Card for Target Matrix**

The format specification may start in any column. The word FORMAT is not used.

**Target Matrix Cards**

The program will read the punched data, row by row, according to the format specified for a row in Format Card for Target Matrix.

**Format Card for Data Matrix**

(Omit this card if IFMT = 0; the program will read the data matrix using the format specified in Format Card for Target Matrix.)

**Data Matrix Cards**

The program will read the punched data, row by row, according to the format specified for a row in Format Card for Data Matrix (Format Card for Target Matrix if IFMT = 0).

continued
System Control Cards
JOB Card will be prepared

System Card 1
Columns   Contents
1-37      //STEPNAME EXEC GITNG0,NAME=MATCHFS

System Card 2
Columns   Contents
1-19      //GO.SYSIN DD *

System Card 3
Columns   Contents
1-2       //

Job Deck
JOB Card
System Card 1
System Card 2
Problem Card
Title Card
Format Card for Target Matrix
Target Matrix Cards
Format Card for Data Matrix (optional)
Data Matrix Cards
Blank Card
System Card 3

May be repeated for additional problems
SAMPLE INPUT
To illustrate the use of the program some simple artificial data were contrived. One factor solution describes four tests in terms of two factors, while the other solution shows the four tests in terms of three factors. Both Case I, for simultaneous rotations, and Case II, for target matching are illustrated.

```
// JOB (....
//STEPNAME EXEC GITNGO,NAME=MATCHFS
//GO.SYSIN DD *
4 2 3 0 0
(3F8.3)
0.700 0.100
0.800 0.000
0.100 0.700
0.000 0.800
0.760 0.320 0.500
0.500 0.500 -0.400
0.520 -0.360 0.500
0.500 -0.500 -0.400
```

10/70
### SAMPLE OUTPUT

#### CASE 1  SIMULTANEOUS ROTATIONS

**Transformation for First Matrix**

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.785</td>
<td>-0.619</td>
</tr>
<tr>
<td>2</td>
<td>0.619</td>
<td>0.785</td>
</tr>
</tbody>
</table>

**Transformation for Second Matrix**

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.994</td>
<td>0.059</td>
</tr>
<tr>
<td>2</td>
<td>0.060</td>
<td>-0.998</td>
</tr>
<tr>
<td>3</td>
<td>0.086</td>
<td>0.016</td>
</tr>
</tbody>
</table>

#### Transformation after Rotation

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.611</td>
<td>-0.355</td>
</tr>
<tr>
<td>2</td>
<td>0.628</td>
<td>-0.496</td>
</tr>
<tr>
<td>3</td>
<td>0.512</td>
<td>0.488</td>
</tr>
<tr>
<td>4</td>
<td>0.496</td>
<td>0.628</td>
</tr>
</tbody>
</table>

**Transformation for Second Matrix**

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.818</td>
<td>-0.267</td>
</tr>
<tr>
<td>2</td>
<td>0.492</td>
<td>-0.476</td>
</tr>
<tr>
<td>3</td>
<td>0.539</td>
<td>0.398</td>
</tr>
<tr>
<td>4</td>
<td>0.433</td>
<td>0.522</td>
</tr>
</tbody>
</table>

#### Transformation to Target

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.744</td>
<td>0.662</td>
</tr>
<tr>
<td>2</td>
<td>0.665</td>
<td>-0.747</td>
</tr>
<tr>
<td>3</td>
<td>0.058</td>
<td>0.066</td>
</tr>
</tbody>
</table>

**Coefficient of Congruence**

0.9861 0.9741
COST ESTIMATE

The pricing algorithm for the Educational Testing Service is based on a complex formula involving the amount of core usage, card-processing, tape and dedicated device usage, lines printed, and central processor unit (CPU) elapsed time. Computer costs for the job included in the Sample Input were $2.30.

\[
\text{Charge to user} = \text{computer costs} + \text{postage and handling} + \text{network overhead} \\
= \$2.30 + \$5.00 + \text{network overhead} \\
= \$7.30 + \text{network overhead}
\]

CONTENTS—

pages  Identification & Abstract  1-2
User Instructions  3-4
I/O  5-6
Cost—Contents  7

10/70
DESCRIPTIVE TITLE: Matrix Decomposition for Points of View Analysis

CALLING NAME: MATDEC

INSTALLATION NAME: Office of Data Analysis Research, Educational Testing Service

AUTHOR(S) AND AFFILIATION(S): J. Ferris, Educational Testing Service

LANGUAGE: FORTRAN IV

COMPUTER: IBM 360/65

PROGRAM AVAILABILITY: Deck and listings presently available

CONTACT: Mr. Ernest Anastasio, Off. of Data Anal. Research, Educational Testing Service, Rosedale Road, Princeton, N.J. 08540 Tel.: (609) 921-9000 ext. 2552

FUNCTIONAL ABSTRACT

The purpose of MATDEC is to decompose a rectangular matrix (i.e., a data matrix, X, dimensioned number of variables by number of subjects) into three matrices U, \( \Gamma \), and W, according to Horst's development (Ref. 1, pp. 364-382), where \( \Gamma \) is a diagonal matrix of eigenvalues and U and W contain the corresponding eigenvectors. It is intended as a first step in Tucker and Messick's approach to an individual differences model for multidimensional scaling.

As is, MATDEC will handle up to 100 variables or subjects, whichever is the lesser dimension of the data matrix. The program uses F4STAT (FORTRAN IV Statistical System) and, in particular, the routine SDGEXT to develop the characteristic roots and vectors of the crossproducts matrix. Further information about F4STAT can be obtained from Mr. Van Hassel, Educational Testing Service, (609) 921-9000, ext. 2557.

REFERENCES


USER INSTRUCTIONS

Input Deck

Title Card

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>NV</td>
<td>1-4</td>
<td>Number variables</td>
</tr>
<tr>
<td>N</td>
<td>5-8</td>
<td>Number of subjects</td>
</tr>
<tr>
<td>IRC</td>
<td>9-12</td>
<td>1: input is rowwise</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2: input is columnwise</td>
</tr>
<tr>
<td>IT</td>
<td>13-16</td>
<td>Tape unit containing input</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5: input on cards</td>
</tr>
<tr>
<td>IR</td>
<td>17-20</td>
<td>1: input tape is to be rewound</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0: input tape is not to be rewound</td>
</tr>
</tbody>
</table>

NOTE: Data matrix is dimensioned NV by N.

Variable-Format Card (optional: card input; IT=5)

Standard FORTRAN format card but delete word FORMAT.

Data Cards (optional)

Follow format on Variable-Format Card.

NOTE: If IRC=1 and an entire matrix row (N entries) is on a single Data Card, then the numbers of Data Cards = NV.

System Control Cards

JOB Card will be prepared by Educational Testing Service personnel.

System Card 1

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-36</td>
<td>//STEPNAME EXEC GITNGO,NAME=MATDEC</td>
</tr>
</tbody>
</table>

System Card 2

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-19</td>
<td>//GO.SYSIN DD *</td>
</tr>
</tbody>
</table>

System Card 3

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>//</td>
</tr>
</tbody>
</table>

continued

3/70

3
Job Deck

JOB Card
System Card 1
System Card 2
Title Card

Parameter Card
Variable-Format Card (optional)
Data Cards (optional)
May be repeated for additional programs.
System Card 3

Description of Output

Printed Output: The parameters NV, N, IRC, IT, and IR are printed at the beginning of the job, and the title is printed at the top of each new page. The raw data are printed followed by the cross-products matrix. An abbreviated form of \( \Gamma \) (the diagonal elements only) is followed by U and W as labeled.

Tape Output: In addition, output will be written in binary on tape unit 3 in the following order.

1. the 72-position array TITLE
2. the diagonal elements of \( \Gamma \)

when \( NV \leq N \):

3. the matrix U, rowwise
4. the matrix W, columnwise

or, when \( NV > N \):

3. the matrix W, columnwise
4. the matrix U, rowwise

If there are additional runs, FT03F002 (OS/360 file number; see contact person for further information) etc., will contain the subsequent output blocks. Note that U, \( \Gamma \), and W are written in double precision.
SAMPLE INPUT

//JOBNAME JOB C....
//STEPNAME EXEC GITNGO,NAME=MATDEC
//GO.SYSIN DD *
SAMPLE CASE TITLE LINE
  6  4  1  5
  (4F1.0)
  1649
  5524
  9452
  4267
  3155
  7831
  //
SAMPLE OUTPUT

NO. OF VARIABLES = 6
NO. OF SUBJECTS = 4
IRC = 1
INPUT TAPE = 5
IR = 0 (1 MEANS REWIND)

ROW 1 OF DATA MATRIX
1.00000 6.00000 4.00000 9.00000

ROW 2 OF DATA MATRIX
5.00000 5.00000 2.00000 4.00000

ROW 3 OF DATA MATRIX
9.00000 4.00000 5.00000 2.00000

ROW 4 OF DATA MATRIX
4.00000 2.00000 6.00000 7.00000

ROW 5 OF DATA MATRIX
3.00000 1.00000 5.00000 5.00000

ROW 6 OF DATA MATRIX
7.00000 8.00000 3.00000 1.00000

CROSS-PRODUCTS

1 2 3 4
1 181.00000 134.00000 119.00000 97.00000
2 134.00000 146.00000 95.00000 109.00000
3 119.00000 95.00000 115.00000 124.00000
4 97.00000 109.00000 124.00000 176.00000

EIGENVALUES (GAMMA)

1 2 3 4
1 22.25328
2 9.30737
3 5.82712
4 1.48615

EIGENVECTORS (W)

1 2 3 4
1 0.5394 -0.60364 -0.37722 -0.44969
2 0.49038 -0.25521 0.78853 0.26949
3 0.45743 0.16536 -0.48038 0.72983
4 0.50910 0.73698 0.07187 -0.43876

continued
<table>
<thead>
<tr>
<th>ROW</th>
<th>1 OF MATRIX U</th>
<th>2 OF MATRIX U</th>
<th>3 OF MATRIX U</th>
<th>4 OF MATRIX U</th>
<th>5 OF MATRIX U</th>
<th>6 OF MATRIX U</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.44458</td>
<td>0.55434</td>
<td>0.52843</td>
<td>0.09266</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.36403</td>
<td>-0.10012</td>
<td>0.23738</td>
<td>-0.80504</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.45489</td>
<td>-0.44619</td>
<td>-0.42886</td>
<td>-0.13300</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.42453</td>
<td>0.34661</td>
<td>-0.39660</td>
<td>0.03220</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.31194</td>
<td>0.26276</td>
<td>-0.40941</td>
<td>0.25284</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.43055</td>
<td>-0.54087</td>
<td>0.59443</td>
<td>0.51056</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COST ESTIMATE

The pricing algorithm for the Educational Testing Service is based on a complex formula involving the amount of core usage, card-processing, tape and dedicated device usage, lines printed and central processor unit (CPU) elapsed time. Computer costs for the job included in the Sample Input were $2.13.

Charge to user = computer costs + postage and handling + network overhead

= $2.13 + $5.00 + network overhead

= $7.13 + network overhead

CONTENTSMATDEC

pages
1 Identification & Abstract
3- 4 User Instructions
5- 7 I/O
9 Cost—Contents
Analysis of Change-over Experiments

ZFE-03, ZFE-04

Office of Data Analysis Research
Educational Testing Service (ETS)

G. Beall
formerly of Gillette Razor Co.

V. Halfmann
formerly of ETS

FORTRAN

IBM 360/65

Decks and listings presently available

Mr. Ernest Anastasio, Off. of Data Anal.
Research, Educational Testing Service,
Rosedale Road, Princeton, N.J. 08540
Tel.: (609) 921-9000 ext. 2552

The class of designs covered by this program is latin squares or Youden rectangles (incomplete latin squares). These may be repeated fully or in part. The design may be defective, i.e., certain whole rows may be missing, but no allowance has been made for missing cells, i.e., single observations.

The purpose of this program is primarily to analyze the results for effect of treatment with allowance for carry-over of preceding treatment. There is also direct testing of the significance of carry-over. There is included parallel estimation and testing of significance without allowance for carry-over. The program is self-contained and does not require any external subroutines, such as might be presumed to exist in one form or another at computation centers.

The program, as presently stored, allows analysis for designs up to 40 rows (or blocks). This is limited by the dimensions of the data matrices. It could be readily enough changed to 500 or 1000 if such an experiment were involved. The analysis has been contrived so that such change does not increase the size of the matrix involved in equation solving.

continued
Explanation

The basic equations are,

1. \( y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_k + \epsilon_{ijk} \)

where \( y_{ijk} \) is an observation assumed built of a general level \( \mu \), effect of the \( i \)th row or individual \( \alpha_i \), the \( j \)th period or column \( \beta_j \), the \( k \)th treatment \( \gamma_k \) and extraneous variability \( \epsilon_{ijk} \). This equation obtains for the first column or period when there has been no conditioning period. For the following periods,

2. \( y_{ijkl} = \mu + \alpha_i + \beta_j + \gamma_k + \delta_{l} + \epsilon_{ijkl} \)

where the effect of the \( k \)th treatment in the preceding period is \( \delta_{l} \). For a conditioned experiment Equ. (2) obtains in all columns.

The data actually considered are the differences within rows such as,

\[ \delta_{ijkl} = \delta_{ijl} - \delta_{ijl} = \beta_j - \beta_j' + \gamma_k - \gamma_{k'} + \delta_{l} - \delta_{l'}, \]

\( j' \neq j, \ k' \neq k, \ l' \neq l \).

These differences are then set forth in a matrix. Thus for an unconditioned latin square for which the first line is,

Design: 1 2 4 3
Result: 4 5 7 6

we may consider the two differences,

\[ \begin{align*}
\delta_{111} - \delta_{1221} &= \beta_1 - \beta_2 + \gamma_1 - \gamma_2 - \delta_1 + \epsilon_{111} - \epsilon_{1221} \\
\delta_{1221} - \delta_{1342} &= \beta_2 - \beta_3 + \gamma_2 - \gamma_4 + \delta_1 - \delta_2 + \epsilon_{1221} - \epsilon_{1342}
\end{align*} \]

which results in two lines of the matrix as follows:

\[
\begin{array}{cccccc}
\beta_1 & \beta_2 & \beta_3 & \gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 & \delta_1 & \delta_2 & \delta_3 & \delta_4 & Result \\
+1 & -1 & +1 & -1 & -1 & -1 & -1 & -1 & +1 & +1 & +1 & -1 & -2 \\
\end{array}
\]

Least-squares equations are in the same form. For instance, to get the equation associated with \( \beta_2 \), each line is multiplied by its content in the \( \beta_2 \) column and the product accumulated over all columns. For each set of effects (\( \beta \), \( \gamma \) or \( \delta \)), the last equation is replaced by a condition equation, \( \text{continued} \)
\[ \sum \hat{\beta}_j = \sum \hat{\gamma}_k = \sum \hat{\delta}_l = 0. \]

The analysis without carry-over (SANS DELTA) is gotten by replacing, temporarily, all equations appropriate to \( \delta \) by \( \delta = 0 \).

The analysis without treatment (SANS GAMMA) is obtained by the temporary replacement, \( \phi_k = 0 \).

The residual variability is gotten in several steps. First the variability residual on \( \beta, \gamma \) and \( \delta \) is

\[
3. \quad \sum y_{ijk}^2 + \sum y_{ijkl}^2 - \sum \hat{\beta}_j (\sum y_{ijk} + \sum y_{ijkl}) - \sum \hat{\gamma}_k (\sum y_{ijkl} + \sum y_{ijkl}) - \sum \hat{\delta}_l \sum y_{ijkl}.
\]

Secondly, an estimate of \( \hat{\beta} \) is made by finding from Equ. (1) or (2) the mean of the values

\[
y'_{ijk} = y_{ijk} - \hat{\beta}_j - \hat{\gamma}_k
\]

\[
y'_{ijkl} = y_{ijkl} - \hat{\beta}_j - \hat{\gamma}_k - \hat{\delta}_l.
\]

Thirdly, estimates of \( \hat{\gamma}_i \) are made by finding the mean of the values

\[
y''_{ijk} = y_{ijk} - \hat{\mu}
\]

\[
y''_{ijkl} = y'_{ijkl} - \hat{\mu}.
\]

The residual variability as from equation (3) is then further and finally reduced by

\[
\hat{\mu} \sum y_{ijkl} + \sum \xi_i \sum y_{ijkl}.
\]

Residual variability on the effects of \( \hat{\mu} \), rows and columns only (SANS DELTA \& SANS GAMMA) is gotten by the formula familiar in analysis of variance.
The test of significance for treatments without allowance for carry-over is based on residual variability SANS DELTA less residual variability SANS DELTA & SANS GAMMA. The test with allowance for carry-over is from residual variability on FULL MATRIX less that on SANS GAMMA. The test for carry-over is from residual variability on FULL MATRIX less that on SANS DELTA.

It need only be added that there is incorporated a test on whether the situation is underdetermined. The program counts the number of different row patterns, multiples this number by the number of columns, and checks whether the result exceeds the number of independent parameters to be estimated. In the case of underdetermination, it refuses to analyze. A second type of refusal arises if the simultaneous equations prove insoluble, which may arise if the design is redundant. Finally, if there is no residual freedom, the program will estimate the parameters but declare $F = 0$. 
USER INSTRUCTIONS

Because there are two experimental designs for change-over experiments, two programs have been written. The first, ZFE-03, analyzes the results from a design that did not include a conditioning week. The second, ZFE-04, analyzes the results from a design that did include a conditioning week. In each case the Parameter Card will give the number of treatments, the number of columns of result, the number of columns of treatment patterns, and the number of rows. In the unconditioned experiment the number of columns of results will equal the number of columns of treatment patterns. In the conditioned experiment, the number of columns of treatment patterns will be one greater than the number of columns of results.

Input

Parameter Card

<table>
<thead>
<tr>
<th>Column</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>Total number of treatments possible</td>
</tr>
<tr>
<td>6-10</td>
<td>Number of columns in treatment pattern</td>
</tr>
<tr>
<td>11-15</td>
<td>Number of columns of results</td>
</tr>
<tr>
<td>16-20</td>
<td>Number of rows of treatments and results</td>
</tr>
</tbody>
</table>

Note: For an experiment with a conditioning period, the number of columns in the treatment pattern will be one greater than the number of columns of results. For an unconditioned experiment, the two numbers will be identical.

Row of Design Cards (one for each row)

<table>
<thead>
<tr>
<th>Format</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>615</td>
<td>Statement of treatments</td>
</tr>
</tbody>
</table>

Row of Result Cards (one for each row)

<table>
<thead>
<tr>
<th>Format</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>6F5.0</td>
<td>Statement of results</td>
</tr>
</tbody>
</table>

System Control Cards

JOB Card will be prepared by ETS personnel.
System Card 1
Columns  Contents
1–30    //STEPNAME EXEC GITNGO,NAME=
31–36    ZFE-03 or ZFE-04 (see above)

System Card 2
Columns  Contents
1–19    //GO.SYSIN DD *

System Card 3
Columns  Contents
1–2     //

Job Deck
JOB Card
System Card 1
System Card 2
Input Deck
System Card 3

Output
The design and results are printed out. The least squares estimates for columns and treatments are printed (SANS DELTA) as they are found without consideration of carry-over. Then the estimates for columns, treatments and carries-over are shown (FULL MATRIX). Finally, the analyses for significance are printed. There is a test for treatment, without allowance for carry-over, a test for treatment with allowance, and a test for carry-over. In each case there is shown the variance ratio, F, with its two statements of freedom.
SAMPLE INPUT

```
//
//STEPNAME EXEC   JOB
//GO.SYSIN DD G1TNGO,NAME=ZFE-03

7  4  4  18
1  2  4  7
2  3  5  1
4  5  7  3
5  6  1  4
6  7  2  5
7  1  3  6
2  6  4  3
3  7  5  4
4  1  6  5
5  2  7  6
6  3  1  7
7  4  2  1
1  2  4  7
2  3  5  1
3  4  6  2
4  5  7  3
5  7  2  5
55. 37. 47. 45.
37. 25. 29. 38.
24. 24. 29. 46.
36. 44. 39. 52.
42. 54. 50. 55.
49. 37. 56. 62.
42. 45. 35. 32.
55. 60. 52. 60.
54. 60. 57. 54.
42. 44. 46. 44.
46. 48. 54. 47.
42. 54. 42. 42.
49. 42. 47. 48.
34. 43. 46. 54.
52. 55. 42. 42.
38. 38. 26. 28.
50. 62. 50. 61.
42. 49. 51. 63.
```
### Estimates of Variables

<table>
<thead>
<tr>
<th>SANS DELTA</th>
<th>FULL MATRIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETA</td>
<td>-1.3858</td>
</tr>
<tr>
<td></td>
<td>-0.1037</td>
</tr>
<tr>
<td></td>
<td>1.3409</td>
</tr>
<tr>
<td></td>
<td>2.8304</td>
</tr>
<tr>
<td>GAMMA</td>
<td>-1.9907</td>
</tr>
<tr>
<td></td>
<td>-1.1776</td>
</tr>
<tr>
<td></td>
<td>4.1289</td>
</tr>
<tr>
<td></td>
<td>0.6008</td>
</tr>
<tr>
<td></td>
<td>0.1367</td>
</tr>
<tr>
<td></td>
<td>-1.2811</td>
</tr>
<tr>
<td></td>
<td>0.7846</td>
</tr>
<tr>
<td>DELTA</td>
<td>-0.0</td>
</tr>
<tr>
<td></td>
<td>-0.0</td>
</tr>
<tr>
<td></td>
<td>-0.0</td>
</tr>
<tr>
<td></td>
<td>-0.0</td>
</tr>
<tr>
<td></td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>0.0</td>
</tr>
</tbody>
</table>

### Analysis

#### FACTORS

<table>
<thead>
<tr>
<th>MU ROWS COLUMNS</th>
<th>CONTROL FACTORS PLUS TREATMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONTROL</td>
<td>TREATMENTS</td>
</tr>
</tbody>
</table>

#### RESIDUAL VAR.

| MU ROWS COLUMNS CONTROL | 2188.37 |
| CONTROL FACTORS PLUS TREATMENTS | 1973.20 |

#### F

\[
F = \frac{(2188.37 - 1973.20)}{6.} = 0.82
\]

#### FACTORS

<table>
<thead>
<tr>
<th>MU ROWS COLUMNS CARRIERS-OVER CONTROL</th>
<th>CONTROL FACTORS PLUS CARRIERS-OVER TREATMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONTROL</td>
<td>TREATMENTS</td>
</tr>
</tbody>
</table>

#### RESIDUAL VAR.

| MU ROWS COLUMNS CARRIERS-OVER CONTROL | 2045.65 |
| CONTROL FACTORS PLUS CARRIERS-OVER TREATMENTS | 1874.19 |

#### F

\[
F = \frac{(2045.65 - 1874.19)}{6.} = 0.59
\]

#### FACTORS

<table>
<thead>
<tr>
<th>MU ROWS COLUMNS TREATMENTS CONTROL</th>
<th>CONTROL FACTORS PLUS CARRIERS-OVER TREATMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONTROL</td>
<td>TREATMENTS</td>
</tr>
</tbody>
</table>

#### RESIDUAL VAR.

| MU ROWS COLUMNS TREATMENTS CONTROL | 1973.20 |
| CONTROL FACTORS PLUS CARRIERS-OVER TREATMENTS | 1874.19 |

#### F

\[
F = \frac{(1973.20 - 1874.19)}{6.} = 0.34
\]
COST ESTIMATE
The pricing algorithm for the Educational Testing Service is based on a complex formula involving the amount of core usage, card-processing, tape and dedicated device usage, lines printed and central processor unit (CPU) elapsed time. Computer costs for the job included in the Sample Input were $3.80.

Charge to user = computer costs + postage and handling + network overhead
= $3.80 + $5.00 + network overhead
= $8.80 + network overhead

CONTENTS—ZFE-03, ZFE-04
pages
1–4 Identification & Abstract
5–6 User Instructions
7–8 I/O
9 Cost—Contents
MANOVA performs univariate and multivariate analyses of variance, of covariance, and of regression. It is quite general and, initially at least, runs should be coordinated with the Office of Computation Sciences through the contact person until the user becomes familiar with the preparation of parameter cards involved. It will perform univariate and multivariate analyses of variance with complete factorial designs or incomplete designs, and with or without covariates, discriminant analyses and canonical correlations. An exact solution in either the orthogonal or nonorthogonal case is provided, and options include single- or multiple-degree-of-freedom contrasts in the main effect or interactions, transformation of variables, and orthogonal polynomial contrasts with equally- or unequally-spaced points. Reanalyses may be done with different criteria, covariates, contrasts and models.

Limitations include a total of 40 variables, no more than 100 nonvacant cells, and up to eight factors with a maximum of 20 levels in each. There is no limitation on the number of subjects.
REFERENCES


Clyde, D.J., Cramer, E.M., & Sherin, R., Revised Manova Program (The University of Miami Biometric Laboratory, Coral Gables, Florida, 1967).


USER INSTRUCTIONS

Input

Input to the program consists of a variable number of cards depending upon the options used. The choice of the options is generally indicated by a 1 punch in a particular column of the Problem Card.

The types of cards used, identification, a summary description of contents and order of cards follow.

Title Card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>TITLE</td>
</tr>
<tr>
<td>6-80</td>
<td>Any title; multiple card may be used</td>
</tr>
</tbody>
</table>

Problem Card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>PROB</td>
</tr>
<tr>
<td>6-7</td>
<td>No. of variables (≤ 40)</td>
</tr>
<tr>
<td>8</td>
<td>No. of factors (≤ 8)</td>
</tr>
<tr>
<td>9</td>
<td>n: No. of Variable-Format Cards</td>
</tr>
<tr>
<td></td>
<td>b: standard format</td>
</tr>
<tr>
<td>10</td>
<td>1: variable names supplied</td>
</tr>
<tr>
<td></td>
<td>b: numbers used</td>
</tr>
<tr>
<td>11</td>
<td>1: subset of variables used as criteria</td>
</tr>
<tr>
<td></td>
<td>b: all variables used</td>
</tr>
<tr>
<td>12</td>
<td>1: Significance Test Cards read</td>
</tr>
<tr>
<td></td>
<td>b: standard ordering</td>
</tr>
<tr>
<td>13</td>
<td>1: last effect on Significance Test Cards obtained as residual and tested</td>
</tr>
<tr>
<td></td>
<td>b: no test on residual effect</td>
</tr>
<tr>
<td>14</td>
<td>1: correlation between observed error variables and canonical variables printed</td>
</tr>
<tr>
<td></td>
<td>b: correlation not printed</td>
</tr>
<tr>
<td>15</td>
<td>4: Input data on tape 4</td>
</tr>
<tr>
<td></td>
<td>b: Input data on tape 5</td>
</tr>
</tbody>
</table>

NOTE: Data on tape 5 must have LRL ≤ 70, RCT = 1, Block ≤ 70, Type 1. Tape labels must be bypassed by 7044 operator. (See contact person for further information.)
### Problem Card

**Columns** | **Contents (b represents blank)**
--- | ---
16 | 1: cell means not printed  
b: cell means printed

17 | 1: transformations of data  
b: no transformations

18 | 1: reduced model matrix printed  
b: matrix not printed

19 | 1: correlations among model parameters printed  
b: correlation not printed

20 | 1: reduced model matrix read in (Col. 12 = 1)  
b: matrix not read

21 | 1: mean omitted from model (Col. 12 = 1)  
b: mean included

22 | 1: sum of product matrix printed  
b: matrix not printed

### Variable-Name Card (optional: PROB Col. (10) = 1)

**Columns** | **Contents**
--- | ---
1-10 | Name of first variable

11-20 | Name of second variable

71-80 | Name of eighth variable

Use number of names equal to number of variables.

### Variable-Subset Card (optional: PROB Col. (11) = 1)

**Columns** | **Contents**
--- | ---
1-2 | Number of criteria (right-justified)

3-4 | Number of covariates (right-justified)

5-80 | Of the V variables, punch variable numbers followed by covariates in 2-digit fields. Use additional cards if necessary starting in Col 1.

*continued*
Contrast Cards (entered for every factor in same order as factor codes)

Columns  Contents (b represents blank)
1  letter ID of factor (not W)
2  b:  regular contrasts 
    1:  special contrasts 
    2:  orthogonal polynomial contrasts on following card 
    3:  orthogonal comparison of i + first level minus average of previous levels 
3-4  Levels for this factor (right-justified < 20) 
5-20  b:  no partition of factor 
      If partitioning to be done, punch degrees of freedom for each partition in 2-digit fields. 
21-60  b:  level codes are punched in Data Cards 
       Level cards are punched in 2-digit fields otherwise. 
61-80  Factor name 

Special Contrast Cards (optional: Contrast Card Col. (2) = 1)

Columns  Contents 
1-80  Square matrix with as many rows as levels in factor in 5-digit fields. Each row begins on new card; they must sum to zero. 

(Contrast Card Col. (2) = 2)

Columns  Contents 
1-80  For orthogonal polynomials, specify metric in 5-digit fields. 

Significance Test Cards (optional: PROB Col. (12) = 1)

Columns  Contents 
1-80  Code letters for effects of design separated by commas, followed by period. W denotes nesting. Ex: B,A,AB. 

Variable-Format Cards (optional: PROB Col. (9) = n)

Columns  Contents 
1-80  Variable format for one subject. Include level code for each factor and include parentheses. 

continued
Transformation Card (optional: PROB Col. (17) = 1)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 to</td>
<td>b: no transformation</td>
</tr>
<tr>
<td>number of variables</td>
<td>1: square root transformation</td>
</tr>
<tr>
<td></td>
<td>2: log transformation</td>
</tr>
<tr>
<td></td>
<td>3: arcsin transformation</td>
</tr>
<tr>
<td>4-9</td>
<td>user's transformation; subroutine SPECTR must be changed</td>
</tr>
</tbody>
</table>

Data Cards (Format designated by PROB Col. (9))

Data must include: numbered level code(s) followed by scores. All Data Cards for one subject must be in order. Deck is followed by as many blank cards as there are Data Cards for one subject.

Reduced Model Matrix Cards (optional: PROB Col. (20) = 1)

First row of reduced model matrix punched in 5-column fields. Continue for remainder of rows in reduced model matrix.

Reanalysis Card (optional)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents (b represents blank)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>ANALY</td>
</tr>
<tr>
<td>6</td>
<td>b: all variables used as criteria</td>
</tr>
<tr>
<td></td>
<td>l: subset used as criteria</td>
</tr>
<tr>
<td>7</td>
<td>b: standard or diagonal ordering</td>
</tr>
<tr>
<td></td>
<td>l: Significance Test Cards read</td>
</tr>
<tr>
<td>8</td>
<td>b: previous contrasts used</td>
</tr>
<tr>
<td></td>
<td>l: new contrasts used</td>
</tr>
<tr>
<td>9</td>
<td>b: residual effect not tested</td>
</tr>
<tr>
<td></td>
<td>l: residual effect tested</td>
</tr>
<tr>
<td>10</td>
<td>b: reduced model matrix not read</td>
</tr>
<tr>
<td></td>
<td>l: reduced model matrix read</td>
</tr>
<tr>
<td>11</td>
<td>b: general mean included</td>
</tr>
<tr>
<td></td>
<td>l: general mean omitted</td>
</tr>
<tr>
<td>12</td>
<td>b: print complete statistical analysis</td>
</tr>
<tr>
<td></td>
<td>l: print estimates of effects only</td>
</tr>
</tbody>
</table>

continued
Variable-Subset Cards (optional: ANALY Col. (6) = 1)
Same format as previous Variable-Subset Cards

Contrast and Special Contrast Cards (optional: ANALY Col. (8) = 1)
Same format as previous Contrast Cards

Significance Test Cards (optional: ANALY Col. (7) = 1)
Same format as previous Significance Test Cards

Reduced Model Matrix Cards (optional: ANALY Col. (10) = 1)
Same format as previous Reduced Model Matrix Cards

Finish Card (placed at end of all problems)
Columns  Contents
1-6      FINISH

System Control Cards
JOB Card will be prepared by ETS personnel.

System Card 1
Columns  Contents
1-36     //STEPNAME EXEC GITNGO,NAME=MANOVA

System Card 2
Columns  Contents
1-19     //GO.SYSIN DD *

System Card 3
Columns  Contents
1-2      //

Job Deck
JOB Card
System Card 1
System Card 2
Input Deck
System Card 3

continued
Output

The output is dependent upon the specifications in the input cards. It may include:

1. description and identification of variables and factors
2. means and standard deviations
3. reduced model matrix
4. correlations of effects
5. univariate analysis table (ANOVA)
6. multivariate
   
   (a) error correlations
   (b) estimates of effects tested against the error term
   (c) analysis of regression if there are covariates

7. a page with multivariate and univariate tests for each effect. In the program, sufficient diagnostic output is given to isolate an error. For more detail, consult the complete writeup (Ref. 1).

Restrictions

No. of variables (< 40)
No. of factors (< 8)
No. of levels for each factor (< 20)
No restriction on number of subjects

REFERENCES

Clyde, D. J., Cramer, E. M., & Sherin, R., Revised MANOVA Program (The University of Miami Biometric Laboratory, Coral Gables, Florida, 1967).
SAMPLE INPUT

// JOB C....
// STEPNAMES EXEC GITN0,NAME=MANOVA
// GO,SYSPIN DD
TITLE PROBLEM 1 FROM HALL AND CRAMER MANOVA PROGRAM WRITTEN IN FORTRAN II
TITLE 2X2 FACTORIAL DESIGN WITH 6 CRITERIA
TITLE REANALYZE WITH 1 CRITERION AND 5 COVARIATES
PROB 6211 1 11

A 2
B 2
(211,10X,6F6.0)
115 41 569. 156. 104. 506. 4. 9.
115 42 475. 120. 103. 366. 4. 16.
115 43 691. 83. 82. 813. 4. 16.
115 44 779. 104. 107. 351. 4. 17.
115 45 587. 98. 55. 564. 4. 13.
115 46 841. 129. 71. 519. 5. 5.
115 47 907. 90. 40. 416. 3. 16.
115 48 698. 76. 13. 492. 4. 9.
115 49 848. 132. 85. 459. 5. 5.
115 50 505. 166. 207. 474. 4. -8.
126 51 557. 91. 62. 513. 3. 36.
126 52 648. 114. 52. 416. 4. 16.
126 53 714. 81. 50. 481. 4. 0.
126 54 611. 125. 80. 630. 3. 25.
126 55 713. 84. 57. 471. 4. 4.
126 56 644. 97. 63. 453. 4. 5.
126 57 555. 83. 55. 546. 4. 8.
126 58 535. 125. 85. 364. 3. 24.
126 59 988. 109. 94. 554. 4. 7.
126 60 584. 120. 91. 503. 4. 4.
128 71 935. 72. 67. 623. 4. 12.
128 72 846. 96. 79. 539. 4. 2.
128 73 704. 109. 65. 355. 2. 7.
128 74 953. 142. 67. 432. 3. 22.
128 75 555. 97. 80. 485. 3. 16.
128 76 592. 82. 67. 362. 2. 17.
128 77 529. 85. 50. 657. 2. 14.
128 78 556. 49. 67. 589. 1. 18.
128 79 419. 103. 77. 452. 3. 15.
128 80 598. 78. 59. 397. 4. 0.
129 81 662. 75. 48. 385. 4. 14.
129 82 668. 88. 48. 361. 2. 8.
129 83 519. 110. 48. 391. 3. 11.
129 84 449. 90. 66. 484. 4. 9.
129 85 847. 80. 56. 482. 4. 2.
129 86 589. 64. 44. 337. 3. 15.
129 87 846. 73. 60. 598. 4. 6.
129 88 748. 46. 65. 601. 4. 30.
129 89 763. 135. 92. 480. 2. 12.
129 90 578. 102. 63. 683. 3. -8.

--- Blank Card ---
SAMPLE OUTPUT

MULTIVARIATE ANALYSIS OF VARIANCE
DATED 3/21/68

PRORFM 1 FROM HALL AND CRAMER MANOVA PROGRAM WRITTEN IN FORTRAN II
2 x 2 FACTORIAL DESIGN WITH 6 CRITERIA
4 ANALYZE WITH 1 CRITERION AND 3 COVARIATES

PROBLEM 1 6 VARIABLES 2 FACTORS

<table>
<thead>
<tr>
<th></th>
<th>ONE</th>
<th>TWO</th>
<th>THREE</th>
<th>FOUR</th>
<th>FIVE</th>
<th>SIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRITERIA</td>
<td>6</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>COVARIATES</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>VARIABLES</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>

FACTOR A 2 LEVELS ALPHA
EVIATION CONTRASTS

FACTOR B 2 LEVELS BETA
EVIATION CONTRASTS

FORMAT OF DATA CARDS
1711111111.

4 CELLS

MEANS AND STANDARD DEVIATIONS

<table>
<thead>
<tr>
<th>FACTOR</th>
<th>A</th>
<th>B</th>
<th>ONE OBS</th>
<th>VARIABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

COMPLETE FACTORIAL WITH NO MISSING CELLS

REDUCED MODEL MATRIX

<table>
<thead>
<tr>
<th>FACTOR</th>
<th>A</th>
<th>B</th>
<th>PARAMETER</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1.000</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>1.000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>1.000</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>1.000</td>
</tr>
</tbody>
</table>

CORRELATIONS OF CONTRASTS WITH STANDARD DEVIATIONS OF CONTRASTS DIVIDED BY STANDARD DEVIATIONS OF VARIABLES ON DIAGONAL

<table>
<thead>
<tr>
<th>CONTRAST</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.158</td>
<td>0.158</td>
<td>0.158</td>
<td>0.158</td>
</tr>
<tr>
<td>1</td>
<td>0.000</td>
<td>-0.950</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>3</td>
<td>0.000</td>
<td>-0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>4</td>
<td>0.000</td>
<td>-0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

continued
### Within Cells Correlations of Criteria with Standard Deviations on Diagonal Adjusted for Covariates

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>ONE</th>
<th>TWO</th>
<th>THREE</th>
<th>FOUR</th>
<th>FIVE</th>
<th>SIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONE</td>
<td>0.063</td>
<td>-0.063</td>
<td>22.762</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWO</td>
<td>-0.171</td>
<td>0.231</td>
<td>26.208</td>
<td>0.011</td>
<td>110.142</td>
<td>0.778</td>
</tr>
<tr>
<td>THREE</td>
<td>0.080</td>
<td>-0.122</td>
<td>0.076</td>
<td>0.74</td>
<td>0.440</td>
<td>8.141</td>
</tr>
<tr>
<td>FOUR</td>
<td>0.395</td>
<td>0.129</td>
<td>0.07</td>
<td>0.74</td>
<td>0.440</td>
<td>8.141</td>
</tr>
<tr>
<td>FIVE</td>
<td>-0.97</td>
<td>0.658</td>
<td>-0.261</td>
<td>-0.117</td>
<td>-0.440</td>
<td>8.141</td>
</tr>
<tr>
<td>SIX</td>
<td>-0.197</td>
<td>-0.058</td>
<td>-0.261</td>
<td>-0.117</td>
<td>-0.440</td>
<td>8.141</td>
</tr>
</tbody>
</table>

### Estimates Adjusted for Covariates

<table>
<thead>
<tr>
<th>CONTRAST</th>
<th>ONE</th>
<th>TWO</th>
<th>THREE</th>
<th>FOUR</th>
<th>FIVE</th>
<th>SIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAST</td>
<td>7.150</td>
<td>7.675</td>
<td>8.900</td>
<td>4.500</td>
<td>0.450</td>
<td>0.250</td>
</tr>
<tr>
<td>LAST</td>
<td>-7.150</td>
<td>-7.675</td>
<td>-8.900</td>
<td>-4.500</td>
<td>-0.450</td>
<td>-0.250</td>
</tr>
<tr>
<td>LAST</td>
<td>11.450</td>
<td>4.375</td>
<td>7.350</td>
<td>2.500</td>
<td>-0.050</td>
<td>0.450</td>
</tr>
<tr>
<td>LAST</td>
<td>-11.450</td>
<td>-4.375</td>
<td>-7.350</td>
<td>-2.500</td>
<td>0.050</td>
<td>-0.450</td>
</tr>
<tr>
<td>LAST</td>
<td>1.150</td>
<td>1.875</td>
<td>4.050</td>
<td>-2.450</td>
<td>0.200</td>
<td>1.750</td>
</tr>
<tr>
<td>LAST</td>
<td>-1.150</td>
<td>-1.875</td>
<td>-4.050</td>
<td>2.450</td>
<td>-0.200</td>
<td>1.750</td>
</tr>
</tbody>
</table>

### Test of AB

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>FI (1, 36)</th>
<th>MEAN SQ</th>
<th>P LESS THAN</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONE</td>
<td>0.759</td>
<td>571.394</td>
<td>0.014</td>
<td>0.510</td>
</tr>
<tr>
<td>TWO</td>
<td>1.478</td>
<td>765.624</td>
<td>0.232</td>
<td>0.098</td>
</tr>
<tr>
<td>THREE</td>
<td>3.146</td>
<td>2160.599</td>
<td>0.005</td>
<td>1.052</td>
</tr>
<tr>
<td>FOUR</td>
<td>0.211</td>
<td>249.938</td>
<td>0.887</td>
<td>0.051</td>
</tr>
<tr>
<td>FIVE</td>
<td>0.165</td>
<td>0.160</td>
<td>0.687</td>
<td>0.267</td>
</tr>
<tr>
<td>SIX</td>
<td>0.122</td>
<td>0.100</td>
<td>0.729</td>
<td>0.362</td>
</tr>
</tbody>
</table>

### Test of R

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>FI (1, 36)</th>
<th>MEAN SQ</th>
<th>P LESS THAN</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONE</td>
<td>0.759</td>
<td>571.394</td>
<td>0.014</td>
<td>0.510</td>
</tr>
<tr>
<td>TWO</td>
<td>1.478</td>
<td>765.624</td>
<td>0.232</td>
<td>0.098</td>
</tr>
<tr>
<td>THREE</td>
<td>3.146</td>
<td>2160.599</td>
<td>0.005</td>
<td>1.052</td>
</tr>
<tr>
<td>FOUR</td>
<td>0.211</td>
<td>249.938</td>
<td>0.887</td>
<td>0.051</td>
</tr>
<tr>
<td>FIVE</td>
<td>0.165</td>
<td>0.160</td>
<td>0.687</td>
<td>0.267</td>
</tr>
<tr>
<td>SIX</td>
<td>0.122</td>
<td>0.100</td>
<td>0.729</td>
<td>0.362</td>
</tr>
</tbody>
</table>

### Test of Significance Using Wilks Lambda Criterion and Canonical Correlations

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>FI (1, 36)</th>
<th>MEAN SQ</th>
<th>P LESS THAN</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONE</td>
<td>0.759</td>
<td>571.394</td>
<td>0.014</td>
<td>0.510</td>
</tr>
<tr>
<td>TWO</td>
<td>1.478</td>
<td>765.624</td>
<td>0.232</td>
<td>0.098</td>
</tr>
<tr>
<td>THREE</td>
<td>3.146</td>
<td>2160.599</td>
<td>0.005</td>
<td>1.052</td>
</tr>
<tr>
<td>FOUR</td>
<td>0.211</td>
<td>249.938</td>
<td>0.887</td>
<td>0.051</td>
</tr>
<tr>
<td>FIVE</td>
<td>0.165</td>
<td>0.160</td>
<td>0.687</td>
<td>0.267</td>
</tr>
<tr>
<td>SIX</td>
<td>0.122</td>
<td>0.100</td>
<td>0.729</td>
<td>0.362</td>
</tr>
</tbody>
</table>

### Test of Significance Using Wilks Lambda Criterion and Canonical Correlations
### TEST OF A TESTS OF SIGNIFICANCE USING MILLS LAMBDA CRITERION AND CANONICAL CORRELATIONS

<table>
<thead>
<tr>
<th>TEST OF HYPOTHESIS</th>
<th>F</th>
<th>DF</th>
<th>P</th>
<th>P LESS THAN</th>
<th>K</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 THROUGH 1</td>
<td>3.786</td>
<td>6</td>
<td>0.077</td>
<td>31.000</td>
<td>0.113</td>
</tr>
</tbody>
</table>

### STANDARDIZED DISCRIMINANT FUNCTION COEFFICIENTS

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>F1</th>
<th>F2</th>
<th>P</th>
<th>P LESS THAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONE</td>
<td>5.467</td>
<td>0.047</td>
<td>0.010</td>
<td></td>
</tr>
<tr>
<td>TWO</td>
<td>4.613</td>
<td>0.006</td>
<td>0.018</td>
<td></td>
</tr>
<tr>
<td>THREE</td>
<td>2.437</td>
<td>0.027</td>
<td>0.039</td>
<td></td>
</tr>
<tr>
<td>FOUR</td>
<td>6.660</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>FIVE</td>
<td>3.678</td>
<td>0.057</td>
<td>0.099</td>
<td></td>
</tr>
<tr>
<td>SIX</td>
<td>7.515</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
</tbody>
</table>

### DISCRIMINANT SCORES

<table>
<thead>
<tr>
<th>CONTRAST</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.757</td>
<td></td>
</tr>
</tbody>
</table>

### MULTIVARIATE ANALYSIS OF VARIANCE

<table>
<thead>
<tr>
<th>SOURCE</th>
<th>SS</th>
<th>DF</th>
<th>MS</th>
<th>F</th>
<th>P LESS THAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>WITHIN CELLS</td>
<td>15.493</td>
<td>31</td>
<td>0.498</td>
<td>2.396</td>
<td>0.048</td>
</tr>
<tr>
<td>REGRESSION</td>
<td>2.036</td>
<td>5</td>
<td>0.287</td>
<td>1.256</td>
<td>0.002</td>
</tr>
<tr>
<td>A</td>
<td>5.549</td>
<td>1</td>
<td>5.549</td>
<td>11.374</td>
<td>0.002</td>
</tr>
<tr>
<td>B</td>
<td>0.007</td>
<td>1</td>
<td>0.007</td>
<td>0.095</td>
<td>0.462</td>
</tr>
<tr>
<td>AB</td>
<td>0.014</td>
<td>1</td>
<td>0.014</td>
<td>1.332</td>
<td>0.276</td>
</tr>
</tbody>
</table>

### ESTIMATES ADJUSTED FOR 5 COVARIATES

<table>
<thead>
<tr>
<th>CONTRAST</th>
<th>FIVE</th>
<th>ONE</th>
<th>TWO</th>
<th>THREE</th>
<th>FOUR</th>
<th>SIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LAST</td>
<td>0.417</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>0.417</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LAST</td>
<td>0.052</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>0.052</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>0.128</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LAST</td>
<td></td>
<td></td>
<td></td>
<td>0.128</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>0.128</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### RAW REGRESSION COEFFICIENTS

<table>
<thead>
<tr>
<th>COVARIATES</th>
<th>WITHIN CELLS</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ONE</td>
<td>0.067</td>
<td>0.067</td>
<td>0.067</td>
<td>0.067</td>
<td>0.067</td>
<td>0.067</td>
</tr>
<tr>
<td>TWO</td>
<td>-0.086</td>
<td>-0.086</td>
<td>-0.086</td>
<td>-0.086</td>
<td>-0.086</td>
<td>-0.086</td>
</tr>
<tr>
<td>THREE</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
<td>0.075</td>
</tr>
<tr>
<td>FOUR</td>
<td>-0.060</td>
<td>-0.060</td>
<td>-0.060</td>
<td>-0.060</td>
<td>-0.060</td>
<td>-0.060</td>
</tr>
<tr>
<td>SIX</td>
<td>-0.378</td>
<td>-0.378</td>
<td>-0.378</td>
<td>-0.378</td>
<td>-0.378</td>
<td>-0.378</td>
</tr>
</tbody>
</table>
COST ESTIMATE

The pricing algorithm for the Educational Testing Service is based on a complex formula involving the amount of core usage, card-processing, tape and dedicated device usage, lines printed and central processor unit (CPU) elapsed time. Computer costs for the job included in the Sample Input were $5.28.

Charge to user = computer costs + postage and handling + network overhead

= $5.28 + $5.00 + network overhead

= $10.28 + network overhead

CONTENTS—MANOVA

pages

1–2 Identification & Abstract
3–8 User Instructions
9–12 I/O
13 Cost—Contents
**DESCRIPTIVE TITLE**
Structure-Factor Least-Squares Refinement Program for IBM 7090

**CALLING NAME**
ORFLS-PX

**INSTALLATION NAME**
University of Pittsburgh
The Computer Center

**AUTHOR(S) AND AFFILIATION(S)**
Dr. Ryonosuke Shiono
Crystallography Laboratory
University of Pittsburgh

**LANGUAGE**
FORTRAN II, except SMI, which is in FAP(UMAP).

**COMPUTER**
IBM 7090 with 32K core storage
2-3 magnetic-tape units in addition to system input/output.

**PROGRAM AVAILABILITY**
Proprietary; usage permitted, but program deck or listing not available

**CONTACT**
Dr. Ryonosuke Shiono, Crystallography Laboratory, 300 Thaw Hall, University of Pittsburgh, Pittsburgh, Pa. 15213
Tel.: (412) 621-3500 ext. 7124

**FUNCTIONAL ABSTRACT**
ORFLS-PX is an extensively modified version of the FORTRAN least-squares program by Busing, Martin, and Levy. Their program, performs successive cycles of refinement using the full matrix of the normal equations. The parameters which may be adjusted include several scale factors, an overall temperature factor coefficient, the neutron scattering factors, individual atom multipliers, atomic coordinates, and isotropic or anisotropic individual atom temperature factor coefficients. The parameters to be varied may be specified arbitrarily, and structures of any symmetry may be accommodated. The refinement may be based either on the structure factors or their squares, and the observations may be weighted individually, or the use of unit weights may be specified.¹

**Features of ORFLS-PX**
1. Reflexion-data input may be from cards or from magnetic tape

¹ continued
2. Uses atomic-scattering-factor table of f versus sinθ
3. External scale factors now refer to $F_{obs}$, rather than to $F_{cal}$'s
4. Weighting scheme may be changed or weights modified by a subroutine
5. Limitations on maximum number of atoms, parameters to be used; number of parameters to be refined have been enlarged
6. Isotropic and anisotropic temperature factors for atoms may be used in mixed fashion
7. Anomalous dispersion correction may be made
8. Calculated structure factors, together with the scales $F$, are written on a binary tape, which may be used as an input for the Fourier program or for a new cycle of the least-squares program.
9. Using an output tape as the input, a part of the structure may be added as fixed contribution for a partial-mode calculation, (this either saves time in calculation or extends the limit of number of atoms to be used)
10. Different set of parameters may be refined in successive cycles of full-matrix refinement, when not all the parameters can be refined at the same time

General Description of the Program Setup

To provide more storage space for the parameter refinement, ORFLS-PX is divided into three separate core-load packages: MAIN(1), MAIN(2), and MAIN(3).

MAIN(1) reads the reflexion data on cards or on tape, interpolates the atomic-scattering factors for each kind, and assigns the weight according to the specification. This part may be bypassed if a tape from a previous cycle of calculation is used as the input. Otherwise, the program produces an intermediate output tape.

MAIN(2) reads back the MAIN(1) tape, one reflexion at a time and calculates structure factor and the necessary derivatives for least-squares refinement. The calculated structure factors and scaled $F_{obs}$ are on the output binary tape. This tape is rewound at the end of every cycle so that the data on this tape are the result of latest calculation. This tape may be used as the input for fixed contribution in a partial-mode calculation.

MAIN(3) prints out the correlation matrix and other information on various agreements at the end of all calculation.
Card formats used for reflexion data and atomic parameters and for symmetry operations are identical to many other interconnected programs on the IBM 7090 and IBM 1130 as well as the IBM 1620. Usually, the user has only to punch a few Control Cards to run any one of the programs.

The Data and Parameter Cards are identical to those in the Block-Diagonal Least-Squares programs\(^3\) on the IBM 7090 and 1130, and so the user may easily switch from one program to another.

The reflexion-data file on magnetic tape has, again, the same format between this and Block-Diagonal, E, Fourier and other programs on the IBM 7090 (see Ref. 3), and it may be used interchangeably.

ORFLS-PX is an independent program, as compared to some others incorporated in a big system, permitting one to write a subroutine to fit a particular problem, and replace the dummy variables provided in the program by one's own.

REFERENCES


3. Additional information can be obtained through the contact person.
USER INSTRUCTIONS

Note for Punching Cards for the Michigan System

The input routine used in FORTRAN in the Michigan System has one bad feature insofar as the standard IBM FORTRAN users are concerned. The I/O routine ignores any blanks in numeric field, whereas the IBM version considers these as zero. As a result, if the following zeros are not punched in an integer field, or in a floating-point number without a decimal point, the Michigan program will take the last punched digit (rather than the last column of the specified field) as the last digit of the field. Therefore, care must be taken to insure that all the cards are punched for the system.

Recommendation: Always punch decimal point in a floating-point number field.

Card Format

Title Card

Any Hollerith characters (alphanumeric) may be punched on Cols. 1-72. The information punched here will be printed on the output as heading.

Control Card No. 1 (5IS, 6F8.4)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>JB</td>
<td>5</td>
<td>0: Input reflexion data from cards</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1: Input reflexion data on tape 9. An output tape of the data-reduction program, E program, or that of the least-squares program (this program or block-diagonal program) may be used. The program uses only h, k, l indices, Fobs, a, ID from the input tape, so that this mode may be used when an input tape contains no atomic f's (data-reduction or E-program output) or some change in weight or atomic f's is desired</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2: Input reflexion on cards. Run only the Part I to produce an intermediate output tape (Tape 10) for use in Patterson synthesis</td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>3:</td>
<td></td>
<td>Input reflexion data on tape from a previous least-squares calculation. The tape should be mounted on tape 10. All information on the tape is used without change, the program bypasses Part I, and the Cell-Constant Card, the Atomic-Scattering Cards, the Reflexion-Data Cards, and the remainder of this card are irrelevant.</td>
</tr>
<tr>
<td>JA</td>
<td>10</td>
<td>0: Refine Fobs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1: Refine on ((F_{o\text{bs}})^2). This causes ((F_{o\text{bs}})^2) to be written on tape 10. Do not use this mode if Fobs's are already squared</td>
</tr>
<tr>
<td>JW</td>
<td>15</td>
<td>0: Standard error σ's are punched on the Data Cards and/or they will be assigned by the subroutine MODIFY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1: Hughes' weighting scheme. When specified, (F_{\text{min}}) (not (4F_{\text{min}})) value should be punched as A in Cols. 26–33</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2: Cruickshank's weighting scheme. A, B, C values below should be supplied.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3: (\omega = \frac{1}{\sin\theta/\lambda})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4: Constant weight, (\omega = 1.0)</td>
</tr>
<tr>
<td>JC</td>
<td>20</td>
<td>0: Subroutine MODIFY is not used</td>
</tr>
<tr>
<td>JD</td>
<td>25</td>
<td>1: Enter subroutine MODIFY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0: Normal</td>
</tr>
<tr>
<td>A</td>
<td>26–33</td>
<td>1: Set (\omega = 0.0) for unobserved reflexion with ID = 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(F_{\text{min}},) value for Hughes' or A for Cruickshank's scheme. This may be used for any other purpose in your MODIFY subroutine</td>
</tr>
<tr>
<td>B</td>
<td>34–41</td>
<td>For Cruickshank's B or blank</td>
</tr>
<tr>
<td>C</td>
<td>42–49</td>
<td>For Cruickshank's C or blank</td>
</tr>
<tr>
<td>D</td>
<td>50–57</td>
<td>For use in MODIFY subroutine</td>
</tr>
<tr>
<td>E</td>
<td>58–65</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>66–73</td>
<td></td>
</tr>
</tbody>
</table>

continued
Weight $w$ is taken as $w = 1/(\sigma)^2$. If $\sigma = 0$, then the program sets $w = 0$. What is punched on the Reflexion-Data Cards or calculated according to some scheme and written on the tape is $\sigma$ and not $w$. Fobs values used here are those on the input cards, or on the tape 9, and they are not affected by the scale factors that the user provides.

**Hughes' Scheme** (Ref. 1)

\[
\begin{align*}
\omega &= 1/|F_0|^2, \text{ if } F_0 \geq 4F_{\text{min}}; \\
\omega &= 1/(4F_{\text{min}})^2, \text{ if } F_0 < 4F_{\text{min}}; \\
\omega &= 0, \text{ if } F_0 = 0.
\end{align*}
\]

**Cruickshank's Scheme** (Ref. 2)

\[
\omega = 1/(A + BF_0 + CF_0^2).
\]

**Cell Constant Card** (3F7.3, 4F7.4)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-7</td>
<td>$a$</td>
</tr>
<tr>
<td>8-14</td>
<td>$b$</td>
</tr>
<tr>
<td>15-21</td>
<td>$c$</td>
</tr>
<tr>
<td>22-28</td>
<td>$\cos\alpha$</td>
</tr>
<tr>
<td>29-35</td>
<td>$\cos\beta$</td>
</tr>
<tr>
<td>36-42</td>
<td>$\cos\gamma$</td>
</tr>
<tr>
<td>43-49</td>
<td>$\lambda$ (wavelength used)</td>
</tr>
</tbody>
</table>

$a$, $b$, $c$, and $\lambda$ are in angstrom units.

**Atomic-Scattering-Factor Cards** (7F10.4)

This is the table of $f$ versus $\sin\theta$, eight cards per atomic kind (51 entries of $f$). Atomic $f$'s are punched in increasing order of corresponding $\sin\theta$ value from 0.00 to 1.00 with an increment of 0.02. A 1620 FORTRAN program or similar one on the 1130 computer prepares the cards from the literature value. The order of sets for each atomic kind determines the number to be used on the parameter cards. Up to ten sets. A blank card terminates the atomic $f$ cards.

continued
Reflexion-Data Cards (3I9, 2F9.2, F9.0, I5, 2F6.3)
The output of the 1130 data-processing program is in this format.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-9</td>
<td>h</td>
</tr>
<tr>
<td></td>
<td>10-18</td>
<td>k indices</td>
</tr>
<tr>
<td></td>
<td>19-27</td>
<td>l</td>
</tr>
<tr>
<td></td>
<td>28-36</td>
<td>F obs</td>
</tr>
<tr>
<td></td>
<td>37-45</td>
<td>c, if used, or blank</td>
</tr>
<tr>
<td></td>
<td>53-54</td>
<td>Scale-factor designation. 1-20. If blank, 1.0 is assumed</td>
</tr>
<tr>
<td></td>
<td>59</td>
<td>0: Observed reflexion 1: Unobserved reflexion</td>
</tr>
<tr>
<td>RA</td>
<td>60-65</td>
<td>Extra fields for any desired values if needed</td>
</tr>
<tr>
<td>RB</td>
<td>66-71</td>
<td>The values will be transferred to the input and the output tape of the refinement part and available in the calculation. Blank if not used</td>
</tr>
</tbody>
</table>

A blank card terminates the reflexion data.

Control Card No. 2 (I615)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>NC</td>
<td>5</td>
<td>The number of cycles to be run. If NC = 0, a set of structure factors will be calculated but no refinement will be made</td>
</tr>
<tr>
<td>NA</td>
<td>8-10</td>
<td>The number of atoms in the asymmetric unit, which are used in the parameter cards</td>
</tr>
<tr>
<td>NQ</td>
<td>14-15</td>
<td>The number of scale factors used and appearing in the parameter list</td>
</tr>
<tr>
<td>ITF</td>
<td>20</td>
<td>0: Isotropic temperature factors and anisotropic temperature factors, or mixed 1: Conversion on the parameter cards of isotropic temperature factors to anisotropic mode before refining. If some of the atoms have already anisotropic temperature factors, this will not affect them.</td>
</tr>
</tbody>
</table>

continued

8 4/70
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>NV</td>
<td>23-25</td>
<td>Number of parameters to be varied and refined. If $N_C = 0$ (unrefined structure factors are used), then $NV$ is irrelevant.</td>
</tr>
<tr>
<td>NAS</td>
<td>29-30</td>
<td>Number of anomalous scattering atoms in the parameter cards to apply the correction. The parameter cards for those atoms must be grouped and placed in front of other atoms.</td>
</tr>
<tr>
<td>NPT</td>
<td>35</td>
<td>0: normal calculation 1: Partial-mode calculation. $A_c, B_c$ are taken from the input tape and added to the current calculated values. This assumes that an output tape of previous calculations is mounted on tape 10 for use as input data. Any atom with anomalous dispersion correction should be in the new part and may not be in the input tape.</td>
</tr>
<tr>
<td>NPRN</td>
<td>40</td>
<td>0: Print structure factors for every cycle. 1: Print only the final structure factors and suppress the printout of other cycles.</td>
</tr>
<tr>
<td>NR</td>
<td>45</td>
<td>0: Normal calculation with no parameter-selection change 1: First run of repeat calculations with change in Parameter-Selection Cards. This makes the program go back and read a new set of Control Cards. 2: Second and more run of the above cycle. The program will go back and read more Control Cards 3: Final run of the recycling. The program terminates at the end of this specification.</td>
</tr>
<tr>
<td>NRW</td>
<td>50</td>
<td>0: Normal 1: Copy back the final output results on tape 4 to tape 10. For this, tape 10 must have the protector ring in. This copying takes place only when the final calculation has been run successfully.</td>
</tr>
</tbody>
</table>

continued
ISTP  55  Contents
0: Program will be terminated, if a nonpositive definite temperature factor results.
1: Program will proceed in the above case by resetting the temperature factor to 0.0001 (for isotropic) or the original value (for anisotropic).

Control Card No. 3 (615)

Parameter  Columns  Contents
NF        5      The number of different x-ray form factors. NF = 0 for neutron diffraction
ICENT     10     1: Centrosymmetric
              2: Noncentrosymmetric
NS        14-15  The number of Symmetry Cards. For centrosymmetric structure, NS is equal to one-half the number of equivalent positions. The original x, y, z should be included. (1 ≤ NS ≤ 24)
IFSQ      20     1: Refine (scale x F)
              2: Refine (scale x F)²
IXFE      25     0: No extra tape for ORFFE³
              1: Extra tape for ORFFE. This may not be specified if the input tape 9 is used.
              2: Punch the same on cards for ORFFE.

Symmetry Cards (F11.6, 212, F11.6, 212, F11.6, 212)
For a centrosymmetric structure, those positions related by the center of symmetry are not required. The basic x, y, z position must be included. For a centered-lattice, those symmetries related by pure translations are not required, but the atom multiplier should be increased.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>blank</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>x</td>
<td>1</td>
</tr>
<tr>
<td>-x</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>y</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>-y</td>
<td>-2</td>
<td></td>
</tr>
<tr>
<td>z</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>-z</td>
<td>-3</td>
<td></td>
</tr>
</tbody>
</table>

continued
### Parameters and Contents

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-11</td>
<td>translational part of x</td>
</tr>
<tr>
<td></td>
<td>12-13</td>
<td>symmetry for x direction</td>
</tr>
<tr>
<td></td>
<td>14-15</td>
<td></td>
</tr>
<tr>
<td></td>
<td>16-26</td>
<td>translational part of y</td>
</tr>
<tr>
<td></td>
<td>27-28</td>
<td>symmetry for y direction</td>
</tr>
<tr>
<td></td>
<td>29-30</td>
<td></td>
</tr>
<tr>
<td></td>
<td>31-41</td>
<td>translational part of z</td>
</tr>
<tr>
<td></td>
<td>42-43</td>
<td>symmetry for z direction</td>
</tr>
<tr>
<td></td>
<td>44-45</td>
<td></td>
</tr>
</tbody>
</table>

### Scale-Factor Cards (8F9.6)

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-9</td>
<td>first scale factor $S_1$</td>
</tr>
<tr>
<td></td>
<td>10-18</td>
<td>second scale factor $S_2$</td>
</tr>
</tbody>
</table>

If there are more than eight scale factors, continue to additional cards. All scale factors in this program apply to the $F_{obs}$ value, rather than to the $F_{cal}$ values. When an output tape of a previous calculation is used as the input, $F_{obs}$ values were already scaled so that the scale factors used here should be 1.0.

### Atomic-Parameter Cards

a) First Card (A6, 3X, 5F9.6, 2F6.2)

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-6</td>
<td>Any 6 Hollerith characters identifying atom $i$</td>
</tr>
<tr>
<td></td>
<td>10-18</td>
<td>X-ray scattering factor identifier, to specify which atomic $f$ to use. (1.0 - 10.0) For neutron problem, this is the neutron-scattering factor itself.</td>
</tr>
<tr>
<td></td>
<td>19-27</td>
<td>A multiplier, $a_i$, this number is usually 1.0, except for a special position or disordered atom, or for a centered-lattice.</td>
</tr>
<tr>
<td></td>
<td>28-36</td>
<td>$x_i$, fractional coordinate</td>
</tr>
<tr>
<td></td>
<td>37-45</td>
<td>$y_i$</td>
</tr>
<tr>
<td></td>
<td>46-54</td>
<td>$z_i$</td>
</tr>
</tbody>
</table>

continued
Columns | Contents
--- | ---
55-60 | Δf' if anomalous dispersion correction is applied, the values are taken as Δf'' constant for sinθ value.

b) Second Card (6F9.6)

Columns | Contents
--- | ---
1-9 | Ti, the isotropic temperature factor, or β11
10-18 | β22, (if the field for β22 is zero, the program assumes isotropic temperature factor
19-27 | β33
28-36 | β12
37-45 | β13
46-54 | β23

The above Atomic Parameter Cards may be prepared from the SFII Coordinate Cards (7070 or 1620) by a 1620 program, and also may be converted back to SFII form. If anomalous dispersion correction is applied for some atoms, the Parameter Cards for those atoms should be placed first.

Parameter-Selection Cards (7211)

These cards specify the NV parameters to be varied, each column of the cards for each parameter, in the order,

Columns | Contents
--- | ---
1 | NQ scale factors
2 | scattering factor or scattering factor multiplier
3 | xi
4 | yi
5 | zi
6 | Bi
7 | β11
8 | β22
9 | β33
10 | β12

continued
<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>$\beta_{13}$</td>
</tr>
<tr>
<td>11</td>
<td>$\beta_{23}$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Punch 0 for holding constant
1 for refinement

The set of six or 11 columns repeat for all atoms used. Continue from Col. 72 of one card to Col. 1 of a next card. At present the total number of parameters to be used including those not to be varied is 811. This card is not used if the number of cycle is zero (unrefined structure factors only).

The number of parameters to be specified for each atom is either six or 11, depending on whether the temperature factor of an atom is isotropic or anisotropic. When using mixed mode of temperature factors, that is, some atoms have isotropic and others have anisotropic temperature factors, care must be taken to count and fill up the columns properly for six or 11. There should be no gap between atoms, and the order of atoms here are identical to the order in which the atomic Parameter Cards are read.

Please note that these Parameter Selection Cards are different from the corresponding ones used for the Block-Diagonal Least-Squares programs\(^3\) (7090, 1130) by Shiono.

**Assembly of Cards**

- 7090 ID Card (Name, time estimate, etc.)
- $\$\$ Comment card(s) to specify scratch or private tapes on 10 and 4.
- $\$NEEDS 10
- $\$HALT (If a private tape is used)
- $\$REMOVE TAPES (If a private tape is used with ring on)
- $\$EXECUTE, I/O DUMP
- Program deck (binary cards)
- $\$DATA 
- Title Card
- Control Card No. 1
- Cell Constant Card

*continued*
Atomic Scattering-Factor Cards (Set of eight, as many sets as necessary up to ten. The order of f's corresponds to the specification on parameter cards.)

Blank Card (to terminate the atomic f's)

Reflexion-Data Cards (as many as necessary)

Blank Card (to terminate the reflexions)

Control Card No. 2

Control Card No. 3

Symmetry Card(s)

Scale Factor Card(s)

Atomic-Parameter Cards (pair of cards for each atom)

Parameter-Selection Card(s) (not used if unrefined structure factors desired)

If JB = 3, Control Card No. 2 follows immediately after Control Card No. 1, and the cards between them are not used.

If JB = 1, omit the Reflexion-Data Cards and a blank card behind them. Control Card No. 2 follows the blank card behind the atomic f's.

If JB = 1, mount the input tape on Tape 9, and a scratch tape on Tape 10. If JB = 3, mount the input tape on Tape 10. If JB = 2, the output is on Tape 10.

The calculated results will be written on Tape 4, and this may be used as an input to the Fourier program or as an input to a later cycle of least-squares program.

When starting with the reflexion on cards, tapes 10 and 4 must have their protector rings on.

Refinement Cycles With Different Set of Parameters Varied

Since the total number of parameters to be varied and refined at a time is limited (at present 181) by the size of core storage, sometimes it is necessary to refine a part of the total structure in a cycle or two and then switch the part to be refined until all the desired parameters have been refined. This may be done by punching a 1, 2 or 3 in Col. 45 of the Control Card No. 2.

Example of a setup of cards is shown below.

continued
Program deck

$DATA

First set of Control Cards for the first selection, 1 in Control Card No. 2, Col. 45.

For second or more refinement cycles,

Title Card
Control Card No. 1  JB = 3 (Col. 5)
Control Card No. 2  NR = 2 (Col. 48)
Control Card No. 3
Symmetry Cards
Parameter Selection Cards

Repeat above set of cards as many as necessary. For the final set of parameter selection (this may follow the first set immediately),

Title Card
Control Card No. 1  JB = 3 (Col. 5)
Control Card No. 2  NR = 3 (Col. 45)
Control Card No. 3
Symmetry Cards
Parameter Selection Cards

Symmetry Cards are usually redundant, but have to be present each time.

If certain part of the total structure is not going to be refined in all cycles, it will be faster to have this part calculated first and use the output tape as the input for fixed contribution.

Program Deck Makeup

MAIN(1)
SCAT
SBTP
MODIFY (Dummy)
BREAK CARD
MAIN(2)
CALC
TEST
SMI
SBTOP
PTOSB
DTOD

continued
A BREAK CARD is recognized as a card with 12, 7 and 9 punches on Col. 1 and rest of the columns all blank up to Col. 72.

Magnetic Tape Assignment

Logical tape addresses used in the program as compiled, are as follows.

<table>
<thead>
<tr>
<th>Symbolic</th>
<th>Logical</th>
<th>Physical</th>
</tr>
</thead>
<tbody>
<tr>
<td>NTPA</td>
<td>7</td>
<td>B1</td>
</tr>
<tr>
<td>NTPB</td>
<td>6</td>
<td>A3</td>
</tr>
<tr>
<td>NTPF</td>
<td>5</td>
<td>A3</td>
</tr>
<tr>
<td>NTPG</td>
<td>10</td>
<td>A5</td>
</tr>
<tr>
<td>NTPC</td>
<td>9</td>
<td>B5</td>
</tr>
<tr>
<td>NTPE</td>
<td>4</td>
<td>A4</td>
</tr>
</tbody>
</table>

Michigan System Input

Michigan System Output (print)

Michigan System Output (card)

Output of Part I, Input of Par. II

Tape input data to Part I (if used) or output tape of parameters to be used in ORFFE.

Output of Part II (identical format of NTPG)

Specification of the Subroutines for the User

1. MODIFY Subroutine

This subroutine is entered, if so specified, in the Part I, to modify reflexion data. After reading a reflexion from card or tape, the main program calculates the Bragg angle and the weight according to the specification and enters this subroutine, just before writing the reflexion on another tape.

```
SUBROUTINE MODIFY (FH,FK,FL,FOBS, FCAL, AΦ,AC,BΦ,BC,SIG, Q,ID,SINT,CA,CB,CC,CD,CE,CF,RA,RB)
```

Parameters     Contents

FH,FK,FL      Miller indices h,k,\( \ell \)

continued
Parameters | Contents
--- | ---
FOBS | observed F (or F^2), punched on card or from Tape 9
FCAL | calculated F
AΦ | observed F, real part
AC | calculated F, real part
BΦ | observed F, imaginary part
BC | calculated F, imaginary part
SIG | standard error of F
Q | designation for scale factor
ID | 0 or 1 for observed or unobserved
SINT | sine theta (Bragg angle)
CA, CB, etc. | A, B, C, D, E, F punched on the Control Card
RA, RB | Extra values on reflexion data on tape. These values may be entered from the input card. Partial difference Fourier requires A_cal, B_cal (partial) be transferred to RA, RB here. (see Fourier writeup)

The values such as FCAL, AΦ, AC, BΦ, BC are meaningful only when the input is the output tape of a previous structure factor (or least-squares) calculation.

In order to exclude a reflexion from the least-squares refinement, SIG should be set to 0.0, which will in turn set \( \omega \) (weight) = 0. Should a reflexion be omitted entirely from the rest of the processing, set ID = 10.

The following three subroutines are explained in Ref. 4 and the user should consult that report. Only the dimension of some arrays have been changed in the current program and they are listed here.

**SUBROUTINE PATCH (I, TJ, HJ, HHJ)**
DIMENSION HJ(3), HHJ(6)

**SUBROUTINE RESETX (XYZ)**
DIMENSION XYZ (3, 85)

**SUBROUTINE RESETB (BETA)**
DIMENSION BETA (6, 85)
1. **RSETW Subroutine**

This subroutine is entered for each reflexion after the completion of the structure factor calculation and before the weight is multiplied to various values in derivative calculation.

```plaintext
SUBROUTINE RSETW (FOBS, FCAL, A, B, SIGYO, SINT, ID)
RETURN
END
```

- **FOBS**: $F_{\text{obs}}$ value already scaled
- **FCAL**: $F_{\text{cal}}$ (absolute)
- **A, B**: A, B part of $F_{\text{cal}}$ with signs
- **SIGYO**: $\sigma$, for zero weight, set $\sigma = 0$
- **SINT**: sin$\theta$ value
- **ID**: 0 for observed and 1 for unobserved

Any changes made here will not affect the contents of Input Tape 10 unless it is so specified to copy back the final Output Tape 4 to Tape 10.

When the user is supplying his subroutine to replace a dummy subroutine in the binary deck, this may be done in two ways,

(a) Compile and punch out the object deck of the subroutine first, and then place this binary deck in front of the dummy subroutine which is going to be replaced, but not before the BREAK Card preceding that part of the program. The system will use the first subroutine loaded.

(b) You may compile a subroutine and execute the entire program at the same time. For this, place the source deck instead of the binary deck of the subroutine as in (a), with a card `$\text{COMPILE FORTRAN, PUNCH OBJECT, PRINT OBJECT}$ preceding it. The system first compiles the subroutine and then executes the entire program.

In either case, the cards for MODIFY should be placed in front of the Binary Deck, and those for PATCH, RESETX, RESE7X or RSETW should be placed immediately behind the first BREAK Card. If the compilation is done with the execution, Tape 9 is needed in compilation and therefore, cannot be used as the input tape.

**Atoms in Special Positions**

Atoms in special positions have to be handled differently from other atoms in general positions. Three subroutine entries are
provided to cope with the situation, if necessary, and they are PATCH, RESEX, and RESEB. The dummy subroutine provided with the deck may be replaced by a user's subroutine, by placing the source statement cards preceded by a $COMPILE FORTRAN Card, or the compiled Binary Cards in front of the one to be replaced.

The detailed instructions for handling special positions are given in Ref. 4, pp. 14-22.

**Reflexion-Tape Record Format**

The program uses the identical record format for the intermediate-input-reflexion data and the output of structure-factor calculation. This format is further identical to that of the Fourier Input Tape, so that either the intermediate tape (Tape 10) or the final output (Tape 4) may be used as an input of the Fourier program3 for calculating Patterson summation or Fourier summation, respectively.

The tape is written in binary mode, and the first record is different from the rest of the data.

**First Record (13 words)**

\[
\frac{1}{4}a^*a^2, \frac{1}{4}b^*b^2, \frac{1}{4}c^*c^2, \frac{1}{4}a^*b^*\cos{\gamma}^*, \frac{1}{4}a^*c^*\cos{\alpha}^*, \frac{1}{4}b^*c^*\cos{\alpha}^*, a, b, c, \cos{\alpha}, \cos{\beta}, \cos{\gamma}, \lambda.
\]

**Second to the (last-1) Record (26 words)**

\[
h, k, \ell, F_{\text{obs}}, F_{\text{cal}}, A_{\text{obs}}, A_{\text{cal}}, B_{\text{obs}}, B_{\text{cal}}, \sigma, Q, ID, (\sin{\theta}/\lambda)^2, \sin{\theta}, (f_i, i=1, 10), RA, RB.
\]

**Parameters**

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>h,k,\ell</td>
<td>Miller indices</td>
</tr>
<tr>
<td>\sigma</td>
<td>Standard error</td>
</tr>
<tr>
<td>Q</td>
<td>Scale-factor identifier</td>
</tr>
<tr>
<td>ID</td>
<td>0 or 1 for observed or unobserved</td>
</tr>
<tr>
<td>RA,RB</td>
<td>Extra values</td>
</tr>
</tbody>
</table>

Only ID is in fixed-point number (integer).

**Last Record**

Identical to the preceding reflexion data except \(h = 999\). This is used to recognize the end of data.

continued
**Programmed Termination**

In several points of the program execution, the run will be terminated before completion when certain conditions are not met.

1. **Parameter Selection Cards Error**
   If the number of parameters to be refined does not match the sum of 1's punched on the Parameter Selection Cards, the program will exit.

2. **Nonpositive-Definite Temperature Factor**
   If a temperature factor becomes nonpositive-definite after applying a shift, the program will terminate any further refinement (Reason 1).

3. **No Improvement in Refinement**
   If $Ew(ΔI)^2/(\text{No. of observation-\text{No. of Parameters}})$ is not decreasing from the previous cycle, the refinement will be terminated at the end of current cycle (Reason 2).

4. **Singular Matrix**
   If the matrix to solve the shifts is a singular matrix, the refinement will be terminated. When this happens, the matrix elements printed may be partially inverted.

**Limitations**

The program deck as compiled has the following capacity for various variables.

<table>
<thead>
<tr>
<th>Variables</th>
<th>Maximum Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of reflexions</td>
<td>no limit</td>
</tr>
<tr>
<td>(one magnetic tape)</td>
<td></td>
</tr>
<tr>
<td>Number of atoms</td>
<td>85</td>
</tr>
<tr>
<td>Number of scale factors</td>
<td>20</td>
</tr>
<tr>
<td>Number of total parameters</td>
<td>811</td>
</tr>
<tr>
<td>Number of parameters to be refined</td>
<td>181</td>
</tr>
<tr>
<td>Number of symmetry relations</td>
<td>24</td>
</tr>
<tr>
<td>Number of different kind of atoms</td>
<td>10</td>
</tr>
<tr>
<td>Number of anomalous dispersing atoms</td>
<td>10</td>
</tr>
</tbody>
</table>

The part 2 of the program has some locations available for the use of user's subroutines PATCH, RESETX, RESETB and RSETW. 

*continued*
When a structure has some variables beyond the limit of the numbers listed above, it may be possible to change the program to handle the structure by recompiling the program with changed dimension statements.

With the partial mode, unlimited number of atoms may be used with multiple passes.

**Output**

I. Card

New scale factors, and the atomic parameters will be punched out for every cycle of refinement. These cards are in the format identical to the input, and may be used in further refinement. If so specified, the input cards for ORFFE may be punched.

II. Print

All information on the two Control Cards (Nos. 2 and 3) as well as the Parameter Cards will be printed first. For each cycle, all structure factors, the agreements and the old and new parameters will be printed. Also one more set of structure factors will be calculated and printed out using the last parameters. If intermediate printing is suppressed, structure factors are printed only from the last calculation.

At the end, the correlation matrix for the parameter, agreement indices for various groups of reflexions and average \((\Delta F/k\sigma)²\) will be printed out.

**NOTE:** If a private tape is used for Tape 10 and/or 4, a card with $HALT$ punched in Cols. 1-5 should be inserted immediately behind the $$ Card(s) for comment. This will halt the program and enable operator to mount the tape.

**REFERENCES**


3. Additional information may be obtained through the contact person.  

continued

## SAMPLE INPUT

<table>
<thead>
<tr>
<th>% DATA</th>
<th>TEST STRUCTURE OF U MO C-i</th>
<th>ACTA, 17, 272 (1964)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.625</td>
<td>5.269 10.980</td>
<td>0 1 0 0 20</td>
</tr>
<tr>
<td>92.000</td>
<td>91.5488 90.2096</td>
<td>88.2013 85.7514</td>
</tr>
<tr>
<td>77.6645</td>
<td>75.0286 72.0012</td>
<td>70.0076 68.9799</td>
</tr>
<tr>
<td>61.4569</td>
<td>59.5136 57.0056</td>
<td>55.9006 54.2126</td>
</tr>
<tr>
<td>49.569</td>
<td>48.1539 46.8149</td>
<td>45.5283 44.2965</td>
</tr>
<tr>
<td>40.8910</td>
<td>39.8500 38.8223</td>
<td>37.8948 36.8086</td>
</tr>
<tr>
<td>34.101</td>
<td>33.6190 32.8575</td>
<td>32.1239 31.4164</td>
</tr>
<tr>
<td>29.435</td>
<td>28.8180 28.2207</td>
<td>27.6434 27.0857</td>
</tr>
<tr>
<td>25.5274</td>
<td>25.0402 0.0000</td>
<td>0.0000 0.0000</td>
</tr>
<tr>
<td>42.0000</td>
<td>41.7204 40.8910</td>
<td>39.6552 38.1626</td>
</tr>
<tr>
<td>33.3890</td>
<td>31.9567 30.8142</td>
<td>29.3460 28.1393</td>
</tr>
<tr>
<td>24.8729</td>
<td>23.9170 23.0234</td>
<td>22.1838 21.3883</td>
</tr>
<tr>
<td>19.2227</td>
<td>18.5804 17.9743</td>
<td>17.4005 16.8556</td>
</tr>
<tr>
<td>12.6144</td>
<td>12.2868 11.9758</td>
<td>11.6782 11.3926</td>
</tr>
<tr>
<td>9.0448</td>
<td>8.6534 .0000</td>
<td>0.0000 0.0000</td>
</tr>
<tr>
<td>6.0000</td>
<td>5.9888 5.9901</td>
<td>5.8578 5.7461</td>
</tr>
<tr>
<td>5.2761</td>
<td>5.0893 5.0883</td>
<td>4.8849 4.7854</td>
</tr>
<tr>
<td>3.8775</td>
<td>3.6885 3.5109</td>
<td>3.3448 3.1883</td>
</tr>
<tr>
<td>2.7771</td>
<td>2.6594 2.5911</td>
<td>2.4535 2.3648</td>
</tr>
<tr>
<td>2.2432</td>
<td>2.1617 2.0258</td>
<td>1.9749 1.8797</td>
</tr>
<tr>
<td>1.8103</td>
<td>1.7778 1.7481</td>
<td>1.7209 1.6952</td>
</tr>
<tr>
<td>1.6282</td>
<td>1.6062 1.5871</td>
<td>1.5694 1.5522</td>
</tr>
<tr>
<td>1.5006</td>
<td>1.4892 .0000</td>
<td>0.0000 0.0000</td>
</tr>
<tr>
<td>0</td>
<td>0        2.26</td>
<td>1 1</td>
</tr>
<tr>
<td>0</td>
<td>0        4.278</td>
<td>1 0</td>
</tr>
<tr>
<td>0</td>
<td>0        8.181</td>
<td>1 0</td>
</tr>
<tr>
<td>1</td>
<td>1        1.133</td>
<td>1 0</td>
</tr>
<tr>
<td>1</td>
<td>1        3.333</td>
<td>1 0</td>
</tr>
<tr>
<td>0</td>
<td>0        1.49</td>
<td>1 0</td>
</tr>
<tr>
<td>1</td>
<td>2        2.53</td>
<td>1 0</td>
</tr>
<tr>
<td>1</td>
<td>2        3.154</td>
<td>1 0</td>
</tr>
<tr>
<td>0</td>
<td>0        0.178</td>
<td>1 0</td>
</tr>
<tr>
<td>0</td>
<td>0        1.210</td>
<td>1 0</td>
</tr>
<tr>
<td>0</td>
<td>0        2.25</td>
<td>1 1</td>
</tr>
<tr>
<td>2</td>
<td>0        0.124</td>
<td>1 0</td>
</tr>
<tr>
<td>2</td>
<td>1        1.68</td>
<td>1 0</td>
</tr>
<tr>
<td>2</td>
<td>1        2.80</td>
<td>1 0</td>
</tr>
<tr>
<td>2</td>
<td>2        0.134</td>
<td>1 0</td>
</tr>
<tr>
<td>2</td>
<td>2        1.170</td>
<td>1 0</td>
</tr>
<tr>
<td>3</td>
<td>0        2.122</td>
<td>1 0</td>
</tr>
<tr>
<td>3</td>
<td>0        12.183</td>
<td>1 0</td>
</tr>
<tr>
<td>3</td>
<td>1        11.79</td>
<td>1 0</td>
</tr>
<tr>
<td>3</td>
<td>3        1.151</td>
<td>1 0</td>
</tr>
<tr>
<td>4</td>
<td>0        1.182</td>
<td>1 0</td>
</tr>
<tr>
<td>4</td>
<td>1        10.155</td>
<td>1 0</td>
</tr>
<tr>
<td>5</td>
<td>1        2.192</td>
<td>1 0</td>
</tr>
<tr>
<td>0</td>
<td>1        2 3</td>
<td>1 0</td>
</tr>
<tr>
<td>0</td>
<td>1        2 3</td>
<td>1 0</td>
</tr>
<tr>
<td>.5</td>
<td>-1       -2 0.5 3</td>
<td>1 0</td>
</tr>
</tbody>
</table>

```
\[
\begin{array}{ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
\end{array}
```

10.0

```
TEST STRUCTURE OF U MO C-2

ACTA. 17, 272 (1964)

A = 5.625, B = 3.249, C = 10.980, COSA = -.0000, COSB = -.0000, COSC = -.0000, LAMBDA = .7093

HUGHES WEIGHTING, FMIN= 20.0

ATOMIC SCATTERING FACTORS

<table>
<thead>
<tr>
<th>Atom</th>
<th>F0</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>B11</th>
<th>B22</th>
<th>B33</th>
<th>B12</th>
<th>B13</th>
<th>B23</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.00</td>
<td>0.0819</td>
<td>1.4211</td>
<td>2.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2.00</td>
<td>0.4191</td>
<td>0.8986</td>
<td>3.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3.00</td>
<td>0.1531</td>
<td>0.7520</td>
<td>6.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>3.00</td>
<td>0.7510</td>
<td>0.0050</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
TEST STRUCTURE OF U MO C-2

ACTA. 17, 272 (1964)

CALCULATED F BASED ON PARAMETERS BEFORE CYCLE 1

<table>
<thead>
<tr>
<th>H</th>
<th>K</th>
<th>L</th>
<th>F(OBS)</th>
<th>F(CALC)</th>
<th>A</th>
<th>B</th>
<th>SINE</th>
<th>SIG(0)</th>
<th>(O-C)/SIG(0)</th>
<th>IQ</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>2</td>
<td>26.0000</td>
<td>20.5575</td>
<td>-17.9495</td>
<td>-10.0212</td>
<td>.065</td>
<td>80.0000</td>
<td>.0680</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>4</td>
<td>278.0000</td>
<td>299.9247</td>
<td>-295.7436</td>
<td>-49.9053</td>
<td>.129</td>
<td>278.0000</td>
<td>-.0789</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>151.0000</td>
<td>159.2828</td>
<td>155.7473</td>
<td>33.2333</td>
<td>.354</td>
<td>151.0000</td>
<td>-.0549</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>162.0000</td>
<td>164.7788</td>
<td>-191.2908</td>
<td>-36.4962</td>
<td>.254</td>
<td>182.0000</td>
<td>-.0702</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>10</td>
<td>155.0000</td>
<td>152.6271</td>
<td>147.1829</td>
<td>40.0007</td>
<td>.424</td>
<td>155.0000</td>
<td>.0153</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>2</td>
<td>192.0000</td>
<td>192.8319</td>
<td>187.8644</td>
<td>43.4071</td>
<td>.340</td>
<td>192.0000</td>
<td>-.0063</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

AGREEMENT FACTORS BASED ON PARAMETERS BEFORE CYCLE 1

SUM(W(0-0)**2) IS 6939E-01

SQRTF(SUM(W(0-0)**2)/(NO-NW)) IS 0.0760

NUMERATOR DENOMINATOR R

R FACTOR INCLUDING UNOBS 154.164 3440.000 0.045
R FACTOR OMITTING UNOBS 146.349 3384.000 0.043
WEIGHTED R FACTOR INCLUDING UNOBS 0.263 4.605 0.057
WEIGHTED R FACTOR OMITTING UNOBS 0.253 4.583 0.055

TEST STRUCTURE OF U MO C-2

ACTA. 17, 272 (1964)

PARAMETERS AFTER LEAST SQUARES CYCLE 1

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>OLD</th>
<th>CHANGE</th>
<th>NEW</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCALE FACTOR</td>
<td>1.0000000</td>
<td>+0037115</td>
<td>1.0037115</td>
<td>+0283736</td>
</tr>
<tr>
<td>U FORM FACTOR</td>
<td>1.0000000</td>
<td></td>
<td>1.0000000</td>
<td></td>
</tr>
<tr>
<td>MULTIPLIER</td>
<td>1.0000000</td>
<td></td>
<td>1.0000000</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>.0819000</td>
<td>-.0012466</td>
<td>.0806533</td>
<td>.0016965</td>
</tr>
<tr>
<td>Y</td>
<td>.2500000</td>
<td></td>
<td>.2500000</td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td>.1421000</td>
<td>-.0040688</td>
<td>.1425068</td>
<td>.0008369</td>
</tr>
<tr>
<td>ATOMIC B</td>
<td>.2000000</td>
<td>-.0092766</td>
<td>.1907234</td>
<td>.1886939</td>
</tr>
<tr>
<td>C 2 FORM FACTOR</td>
<td>3.0000000</td>
<td></td>
<td>3.0000000</td>
<td></td>
</tr>
<tr>
<td>MULTIPLIER</td>
<td>1.0000000</td>
<td></td>
<td>1.0000000</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>.7510000</td>
<td>-.0092766</td>
<td>.7218003</td>
<td>.0225634</td>
</tr>
<tr>
<td>Y</td>
<td>.2500000</td>
<td></td>
<td>.2500000</td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td>.0050000</td>
<td>-.0061485</td>
<td>-.0011485</td>
<td>.0173400</td>
</tr>
<tr>
<td>ATOMIC B</td>
<td>.0000000</td>
<td>1.6818284</td>
<td>1.6818284</td>
<td>1.9679505</td>
</tr>
</tbody>
</table>

ESTIMATED AGREEMENT FACTORS BASED ON PARAMETERS AFTER CYCLE 1

SUM(W(0-0)**2) IS 3529E-01

SQRTF(SUM(W(0-0)**2)/(NO-NW)) IS .0542

continued
### Calculated F Based on Parameters Before Cycle 2

<table>
<thead>
<tr>
<th>H</th>
<th>K</th>
<th>L</th>
<th>F(805)</th>
<th>F(CALC)</th>
<th>A</th>
<th>B</th>
<th>SINE</th>
<th>SIG(0)</th>
<th>SIG0/SIG1</th>
<th>IQ</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>2</td>
<td>26.0965</td>
<td>20.0279</td>
<td>-17.2174</td>
<td>-10.2313</td>
<td>.065</td>
<td>80.0000</td>
<td>.0756</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>4</td>
<td>79.0318</td>
<td>298.0844</td>
<td>-293.9271</td>
<td>-49.6104</td>
<td>.129</td>
<td>276.0000</td>
<td>-.0683</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>6</td>
<td>56.2078</td>
<td>57.9914</td>
<td>50.4084</td>
<td>28.6706</td>
<td>.194</td>
<td>80.0000</td>
<td>-.0222</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>8</td>
<td>103.6718</td>
<td>176.4055</td>
<td>173.8244</td>
<td>33.3538</td>
<td>.258</td>
<td>181.0000</td>
<td>.0257</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>113.4194</td>
<td>111.5944</td>
<td>-110.2726</td>
<td>-17.1248</td>
<td>.130</td>
<td>113.0000</td>
<td>.0161</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>334.2359</td>
<td>314.6329</td>
<td>-311.1163</td>
<td>-46.9083</td>
<td>.142</td>
<td>333.0000</td>
<td>.0597</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>164.5365</td>
<td>145.2431</td>
<td>143.4101</td>
<td>23.0027</td>
<td>.159</td>
<td>144.0000</td>
<td>-.0049</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>49.1819</td>
<td>48.7262</td>
<td>-41.5077</td>
<td>-25.5215</td>
<td>.230</td>
<td>80.0000</td>
<td>.0057</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
<td>51.1967</td>
<td>52.8388</td>
<td>47.8114</td>
<td>22.4942</td>
<td>.247</td>
<td>54.0000</td>
<td>.1082</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>154.5716</td>
<td>137.8471</td>
<td>132.2144</td>
<td>39.0024</td>
<td>.247</td>
<td>154.0000</td>
<td>.1082</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>178.6607</td>
<td>176.7828</td>
<td>174.3810</td>
<td>29.0419</td>
<td>.126</td>
<td>178.0000</td>
<td>.0105</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>210.7794</td>
<td>217.1951</td>
<td>-214.1845</td>
<td>-36.0378</td>
<td>.130</td>
<td>210.0000</td>
<td>-.0304</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>25.9029</td>
<td>25.2177</td>
<td>-24.6224</td>
<td>-5.4356</td>
<td>.143</td>
<td>80.0000</td>
<td>.0016</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>124.4602</td>
<td>129.1477</td>
<td>-122.5902</td>
<td>-40.6296</td>
<td>.167</td>
<td>124.0000</td>
<td>-.0377</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>68.2524</td>
<td>70.5149</td>
<td>-67.5649</td>
<td>-20.2428</td>
<td>.170</td>
<td>80.0000</td>
<td>-.0282</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>80.2969</td>
<td>83.3624</td>
<td>82.7058</td>
<td>10.4424</td>
<td>.179</td>
<td>80.0000</td>
<td>-.0382</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
<td>134.4973</td>
<td>137.5550</td>
<td>-130.4895</td>
<td>-28.4506</td>
<td>.252</td>
<td>134.0000</td>
<td>.0070</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
<td>170.6310</td>
<td>164.8388</td>
<td>160.8063</td>
<td>33.3538</td>
<td>.254</td>
<td>170.0000</td>
<td>.0351</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
<td>122.6528</td>
<td>120.4900</td>
<td>-111.0855</td>
<td>-46.6674</td>
<td>.200</td>
<td>122.0000</td>
<td>.0160</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>183.6792</td>
<td>176.7828</td>
<td>174.3810</td>
<td>29.0419</td>
<td>.126</td>
<td>178.0000</td>
<td>.0105</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>210.7794</td>
<td>217.1951</td>
<td>-214.1845</td>
<td>-36.0378</td>
<td>.130</td>
<td>210.0000</td>
<td>-.0304</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>220.1268</td>
<td>210.1268</td>
<td>212.4191</td>
<td>133.1597</td>
<td>.380</td>
<td>151.0000</td>
<td>.0095</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>188.7555</td>
<td>188.3604</td>
<td>-184.6502</td>
<td>-37.2011</td>
<td>.254</td>
<td>182.0000</td>
<td>-.0311</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0</td>
<td>183.6792</td>
<td>176.7828</td>
<td>174.3810</td>
<td>29.0419</td>
<td>.126</td>
<td>178.0000</td>
<td>.0105</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>2</td>
<td>192.7126</td>
<td>189.7096</td>
<td>184.8685</td>
<td>42.4972</td>
<td>.340</td>
<td>192.0000</td>
<td>.0156</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

### Agreement Factors Based on Parameters Before Cycle 2

SUMW*(O-C)**2)/(NO-NV) = .3582E-01

\[
\text{SQRFSUMW*(O-C)**2} = .0546
\]

<table>
<thead>
<tr>
<th>NUMERATOR</th>
<th>DENOMINATOR</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>R Factor Including Unobs</td>
<td>116.299</td>
<td>3452.768</td>
</tr>
<tr>
<td>R Factor Omitting Unobs</td>
<td>110.105</td>
<td>3401.578</td>
</tr>
<tr>
<td>Weighted R Factor Including Unobs</td>
<td>.189</td>
<td>4.605</td>
</tr>
<tr>
<td>Weighted R Factor Omitting Unobs</td>
<td>.174</td>
<td>4.583</td>
</tr>
</tbody>
</table>

continued
<table>
<thead>
<tr>
<th></th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
<th>1.0000</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.0000</td>
<td>0.0424</td>
<td>-0.1581</td>
<td>0.7396</td>
<td>-0.1126</td>
<td>-0.0477</td>
<td>0.2227</td>
<td>0.0030</td>
<td>-0.1603</td>
<td>0.2436</td>
<td>-0.0826</td>
<td>0.0598</td>
<td>0.2489</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1.0000</td>
<td>0.0655</td>
<td>0.0513</td>
<td>0.1258</td>
<td>-0.0040</td>
<td>-0.1101</td>
<td>0.1467</td>
<td>0.0723</td>
<td>0.0797</td>
<td>0.0907</td>
<td>0.0042</td>
<td>0.1213</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1.0000</td>
<td>-0.1651</td>
<td>-0.0552</td>
<td>0.0273</td>
<td>-0.1087</td>
<td>0.0916</td>
<td>-0.1553</td>
<td>-0.4591</td>
<td>-0.2431</td>
<td>0.3275</td>
<td>0.2424</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1.0000</td>
<td>-0.0635</td>
<td>-0.3191</td>
<td>0.0740</td>
<td>-0.0222</td>
<td>0.0106</td>
<td>0.1986</td>
<td>-0.0746</td>
<td>-0.0870</td>
<td>0.4436</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1.0000</td>
<td>-0.0040</td>
<td>0.0040</td>
<td>0.3686</td>
<td>0.3758</td>
<td>0.0534</td>
<td>0.3955</td>
<td>0.0963</td>
<td>-0.2244</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1.0000</td>
<td>0.3857</td>
<td>0.0936</td>
<td>-0.1463</td>
<td>0.0846</td>
<td>-0.0805</td>
<td>-0.1663</td>
<td>-0.1475</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1.0000</td>
<td>-0.1715</td>
<td>0.0166</td>
<td>0.1372</td>
<td>-0.1361</td>
<td>0.2988</td>
<td>-0.2061</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1.0000</td>
<td>0.0396</td>
<td>0.1236</td>
<td>0.2130</td>
<td>0.2034</td>
<td>0.0466</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1.0000</td>
<td>-0.0286</td>
<td>0.1441</td>
<td>0.4963</td>
<td>0.0912</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1.0000</td>
<td>0.3886</td>
<td>0.0739</td>
<td>0.0410</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>1.0000</td>
<td>-0.0326</td>
<td>-0.1519</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1.0000</td>
<td>-0.0781</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SINE THETA UP TO</td>
<td>0.1</td>
<td>0.2</td>
<td>0.3</td>
<td>0.4</td>
<td>0.5</td>
<td>0.6</td>
<td>0.7</td>
<td>0.8</td>
<td>0.9</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO. REFLEXION</td>
<td>12</td>
<td>7</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R FACTOR</td>
<td>0.233</td>
<td>0.036</td>
<td>0.038</td>
<td>0.013</td>
<td>0.018</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HKL</td>
<td>HK0</td>
<td>OKL</td>
<td>HKL</td>
<td>H0L</td>
<td>H00</td>
<td>OK0</td>
<td>OOL</td>
<td>ALL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO. REFLEXION</td>
<td>13</td>
<td>2</td>
<td>0</td>
<td>5</td>
<td>1</td>
<td>0</td>
<td>4</td>
<td>25</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R FACTOR</td>
<td>0.034</td>
<td>0.022</td>
<td>0.000</td>
<td>0.026</td>
<td>0.011</td>
<td>0.000</td>
<td>0.058</td>
<td>0.034</td>
<td>0.038</td>
<td>0.094</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EEE</td>
<td>EEO</td>
<td>EOE</td>
<td>OEE</td>
<td>E00</td>
<td>O00</td>
<td>O0E</td>
<td>000</td>
<td>000</td>
<td>000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO. REFLEXION</td>
<td>7</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R FACTOR</td>
<td>0.039</td>
<td>0.032</td>
<td>0.025</td>
<td>0.019</td>
<td>0.033</td>
<td>0.084</td>
<td>0.043</td>
<td>0.012</td>
<td>0.010</td>
<td>0.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OBS. ONLY SUM F0BS</td>
<td>3385.20</td>
<td>0.995</td>
<td>51.19</td>
<td>45.25</td>
<td>0.994</td>
<td>0.084</td>
<td>0.043</td>
<td>0.012</td>
<td>0.010</td>
<td>0.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F VALUE UP TO</td>
<td>10.</td>
<td>20.</td>
<td>30.</td>
<td>40.</td>
<td>50.</td>
<td>60.</td>
<td>70.</td>
<td>80.</td>
<td>90.</td>
<td>100.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO. REFLEXION</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AVER. W*DELTSQ</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
<td>0.000</td>
<td>0.002</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO. REFLEXION</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AVER. W*DELTSQ</td>
<td>0.000</td>
<td>0.000</td>
<td>0.003</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SINE THETA UP TO</td>
<td>0.1</td>
<td>0.2</td>
<td>0.3</td>
<td>0.4</td>
<td>0.5</td>
<td>0.6</td>
<td>0.7</td>
<td>0.8</td>
<td>0.9</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO. REFLEXION</td>
<td>0</td>
<td>11</td>
<td>7</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AVER. W*DELTSQ</td>
<td>0.000</td>
<td>0.001</td>
<td>0.002</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO. REFLEXION</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AVER. W*DELTSQ</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COST ESTIMATE
Unfortunately, exact figures are not available for central-processor time. The program listed on the Sample Input used approximately 10.8 minutes of central-processor time. At the current rate of $250/hour at the University of Pittsburgh, the chargeable computer time was approximately $45.00.

Cost to user = computer time + network overhead

= $45.00 (estimated) + network overhead

CONTENTS—ORFLS-PX

pages
1–3 Identification & Abstract
5–22 User Instructions
23–28 I/O
29 Cost—Contents
Factor analysis provides that the final solution be in terms of either uncorrelated factors or correlated factors. Beginning in the mid-1940's, following the leadership of Thurstone, there was a trend toward the acceptance of oblique factors. This trend has continued to the present day but, unfortunately, efficient objective means for getting oblique "simple structure" solutions have not generally been available even with modern computers. In 1958, John B. Carroll introduced a whole class of methods for oblique transformation to simple structure. These have come to be known by the term "oblimin" (Ref. 1, pp. 324-326), since they involve oblique factors and the minimization of a function. The oblimin criterion, which is to be minimized, is given in normalized form by,

\[
B = \sum_{p<q=1}^{n} \left[ \sum_{j=1}^{n} v_{jp}/h_{j} (v_{jq}/h_{j}) - \gamma \sum_{j=1}^{n} v_{jp}/h_{j} \right]^2
\]
The v's in these expressions are the elements of the reference-factor structure matrix, i.e., the correlations between the original variables and the reference factors. Actually, what is desired is to have the primary factor pattern exhibit the principles of simple structure, i.e., large values and near-zero values. In the Thurstone school the "reference structure V" is sought that exhibits the simple structure principles, and then the primary factor pattern is obtained by multiplying the matrix V by a diagonal matrix.

The foregoing indirect, and somewhat awkward, procedure has recently been replaced by a direct approach (Ref. 2). Instead of working with the reference factors that are biorthogonal to the primary factors, Jennrich and Sampson set up a criterion for the direct determination of the primary factors that exhibit the simple structure principles. That criterion may be put in the form,

\[ F(A) = \sum_{p<q=1}^{m} \left( \sum_{j=1}^{n} a_{pj}^2 \right) - \frac{\delta}{n} \left( \sum_{j=1}^{n} a_{j1}^2 \right) \left( \sum_{j=1}^{n} a_{j2}^2 \right) \]

where A is the matrix of primary factor coefficients. Of course, the loadings may be normalized by rows just as in equation 1. An important difference is that the \( \gamma \) in the indirect method ranges between zero and one, while the \( \delta \) in equation 2 should be zero or negative.

The object of OBLIMIN is to minimize equation 2. The criterion employed is,

\[ \frac{F_{i-1} - F_i}{F_0} \leq \varepsilon \]

where \( i \) is the iteration number. The output is an oblique factor solution satisfying the principles of simple structure, more or less. When \( \delta \) is equal to zero, the factors are most oblique. For negative values of \( \delta \), the factors become less oblique as \( \delta \) gets smaller. The solution consists of the factor pattern, the correlations among the factors, and the factor structure. At the present time, the program is limited to \( n = 100 \) variables and \( m = 15 \) factors.

REFERENCES


USER INSTRUCTIONS

System Control Cards

JOB Card will be prepared by ETS personnel.

System Card 1

Columns     Contents
  1-37        ///STEPNAME EXEC GITNGO,NAME=OBLIMIN

System Card 2

Columns     Contents
  1-19        ///GO.SYSIN DD *

System Card 3

Columns     Contents
  1-2         //

Problem Card

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Columns</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>3-5</td>
<td>I3</td>
<td>Number of variables (≤ 100)</td>
</tr>
<tr>
<td>M</td>
<td>6-10</td>
<td>I5</td>
<td>Number of factors (≤ 15)</td>
</tr>
<tr>
<td>DEL</td>
<td>11-15</td>
<td>F5.1</td>
<td>Delta (except for experimental purposes, δ ≤ 0; if several values are used, smallest goes in this field)</td>
</tr>
<tr>
<td>EPS</td>
<td>16-25</td>
<td>F10.6</td>
<td>Convergence criterion (usually = .00001)</td>
</tr>
<tr>
<td>MAXI</td>
<td>26-30</td>
<td>I5</td>
<td>Maximum number of iterations (usually 50)</td>
</tr>
<tr>
<td>IF</td>
<td>31-35</td>
<td>I5</td>
<td>Original factor pattern option: 1 = orthogonal (usual case, requires no additional input) 0 = oblique (correlations among the original factors must be input in addition to the pattern)</td>
</tr>
</tbody>
</table>

6/70       3

continued
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Columns</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>36-40</td>
<td>I5</td>
<td>Kaiser normalization option: 1 = used (usual case) 0 = not used</td>
</tr>
<tr>
<td>DINC</td>
<td>41-45</td>
<td>F5.1</td>
<td>Delta increment (must be a positive number)</td>
</tr>
<tr>
<td>DEND</td>
<td>46-50</td>
<td>F5.1</td>
<td>Largest delta</td>
</tr>
<tr>
<td>NOSTOP</td>
<td>51-55</td>
<td>I5</td>
<td>1 = additional problem(s) to follow 0 = last problem</td>
</tr>
</tbody>
</table>

**NOTE:** The last three fields may be blank if only one value of delta and one problem are run.

**Format Card**

The format specification starts in Col. 2 and may not extend beyond Col. 72. The word FORMAT is not used. The standard FORTRAN variable format is followed.

**Data Cards**

Keypunch original factor pattern by rows according to its format, starting each variable on a new card.

**Factor Correlations Cards**

When the original factor pattern is oblique, a "0" must be put in Col. 35 of the Problem Card, and the following cards must be inserted at this point,

(a) FORMAT Cards (Cols. 2-72)

(b) Correlations among factors, starting a new card for each factor

**Title Card**

Name of problem in Cols. 2-80.

**Order of Cards**

JOB Card
System Card 1
System Card 2
Problem Card
Format Card

*continued*
Data Cards
Factor Correlations Cards
Title Card
/*
System Card 3

Description of Output

See Functional Abstract for a description of the output of OBLIMIN.
SAMPLE INPUT

// JOB (....
//STEPNAME EXEC GITNGO,NAME=OBLIMIN
//GO.SYSIN DD *

(2F10.5)
0.85600 -0.32400
0.84800 -0.41200
0.80800 -0.40900
0.83100 -0.34200
0.75000 0.57100
0.63100 0.49200
0.56900 0.51000
0.60700 0.35100
/*
/*
SAMPLE OUTPUT

DELTA = 0.00

ROTATION FOR DIRECT OBLIMIN LOADINGS

<table>
<thead>
<tr>
<th>ITERATION</th>
<th>DIROBL CRITERION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.905183</td>
</tr>
<tr>
<td>1</td>
<td>2.683994</td>
</tr>
<tr>
<td>2</td>
<td>0.731741</td>
</tr>
<tr>
<td>3</td>
<td>0.072721</td>
</tr>
<tr>
<td>4</td>
<td>0.072715</td>
</tr>
</tbody>
</table>

AFTER ROTATION WITH KAISER NORMALIZATION

FACTOR PATTERN

| 1 | 0.88272 | 0.06520 |
| 2 | 0.95632 | -0.02946 |
| 3 | 0.92599 | -0.04505 |
| 4 | 0.88163 | 0.03492 |
| 5 | 0.00471 | 0.94040 |
| 6 | -0.00647 | 0.80317 |
| 7 | -0.06549 | 0.79280 |
| 8 | 0.10374 | 0.64627 |

FACTOR CORRELATIONS

| 1 | 1.00000 | 0.47149 |
| 2 | 0.47149 | 1.00000 |

FACTOR STRUCTURE

| 1 | 0.91346 | 0.48139 |
| 2 | 0.94243 | 0.42143 |
| 3 | 0.90475 | 0.39154 |
| 4 | 0.89810 | 0.45060 |
| 5 | 0.44809 | 0.94262 |
| 6 | 0.37221 | 0.80012 |
| 7 | 0.30831 | 0.76192 |
| 8 | 0.40845 | 0.69518 |

continued
DELTA = -0.50

ROTATION FOR DIRECT OBLIMIN LOADINGS

DELTA = -1.00

ROTATION FOR DIRECT OBLIMIN LOADINGS

DELTA = -1.50

ROTATION FOR DIRECT OBLIMIN LOADINGS

DELTA = -2.00

ROTATION FOR DIRECT OBLIMIN LOADINGS

ITERATION DIROBL CRITERION
0 9.111930  9 7.357594
1 9.050656 10 7.303166
2 8.985585 11 7.282603
3 8.874997 12 7.275172
4 8.695285 13 7.272531
5 8.426510 14 7.271598
6 8.080988 15 7.271269
7 7.735189 16 7.271153
8 7.487005 17 7.271112

AFTER ROTATION WITH KAISER NORMALIZATION

continued
FACTOR PATTERN

1  0.86186  0.14201
2  0.92372  0.05725
3  0.89278  0.03949
4  0.85778  0.11269
5  0.09845  0.90789
6  0.07392  0.77446
7  0.01569  0.75917
8  0.16504  0.63294

FACTOR CORRELATIONS

1  1.00000  0.30538
2  0.30538  1.00000

FACTOR STRUCTURE

1  0.90522  0.40520
2  0.94121  0.33934
3  0.90484  0.31213
4  0.89219  0.37464
5  0.37570  0.93795
6  0.31043  0.79704
7  0.24753  0.76396
8  0.35833  0.68334
COST ESTIMATE

Exact figures for the program listed on the Sample Input are not available. As an approximate guide to the user, an OBLIMIN run whose parameters lie within the specified range will cost less than $20.00 with a typical cost of less than $10.00.

Approximate cost to user = computer time + postage and handling + network overhead
= $10.00 + $5.00 + network overhead
= $15.00 + network overhead

CONTENTS—OBLIMIN

pages
1–2 Identification & Abstract
3–5 User Instructions
7–10 I/O
11 Cost—Contents
FUNCTIONAL ABSTRACT

UMLFA performs a factor analysis of a given correlation matrix. "The factor loadings and the unique variances are estimated by Lawley's method of maximum likelihood. The computational procedure...makes use of the method of Fletcher and Powell for numerical minimization of a function. Any number of factors can be extracted, and each factor is rotated, using Kaiser's varimax method. The goodness of fit of the maximum likelihood solution is tested by Lawley's chi-square test based on the likelihood ratio technique."¹

The program is able to handle up to 75 variables and 30 factors. However, the input correlation matrix must be positive definite.

REFERENCES


USER INSTRUCTIONS

Specification of Input

360 O.S./JCL cards (job control)

Col. 1    Col. 16

//   JOB
//JOBLIB   DD   DSN=OCS.SAPGM,DISP=SHR
//GO       EXEC   PGM=UMLFA,TIME=2,REGION=150K,ACCT=xxxxx
//FT05F001 DD   SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=3458)
//FT07F001 DD   SYSOUT=B,DCB=(RECFM=FB,LRECL=80,BLKSIZE=3520)
//FTXXF001 DD   SPACE=(CYL,(1,1)),DISP=NEW,UNIT=2314

[Note: Replace XX in above card with number punched in Cols. 29-30 of parameter card no. 2.]

//FT05F001 DD   *
(UMLFA control cards and data go here)

/ *
/

Heading Card

Any information may be punched in Cols. 1-72.

Parameter Card

<table>
<thead>
<tr>
<th>Column</th>
<th>Value or Variable</th>
<th>Description</th>
<th>(Recommended Value)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>P</td>
<td>Number of variables</td>
<td></td>
</tr>
<tr>
<td>6-10</td>
<td>K</td>
<td>Lower bound for number of factors (≥ 1)</td>
<td></td>
</tr>
<tr>
<td>11-15</td>
<td>K</td>
<td>Upper bound for number of factors (≤ 30)</td>
<td></td>
</tr>
<tr>
<td>16-20</td>
<td>NP</td>
<td>Sample size</td>
<td></td>
</tr>
<tr>
<td>21-25</td>
<td>MAXITER</td>
<td>Maximum number of iterations</td>
<td>(safely, 3P)</td>
</tr>
<tr>
<td>26-30</td>
<td></td>
<td>Scratch tape number</td>
<td></td>
</tr>
<tr>
<td>31-35</td>
<td>CRV</td>
<td>Convergence criterion</td>
<td>(usually 1.0)</td>
</tr>
<tr>
<td>41</td>
<td>1/0</td>
<td>Print/don't print original correlation matrix and partial correlation matrices with some variables eliminated</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>1/0</td>
<td>Print/don't print technical output related to the iterative process</td>
<td>(usually 0)</td>
</tr>
</tbody>
</table>

continued
Print/don't print intermediate results when some variables are eliminated, and the reproduced correlations and residuals for the final solution

Punch/don't punch unrotated factor matrix (0, unless it is desired to use the result for input to another rotation program)

Suppress/don't suppress boundary elimination

Nonstandard/standard parameters

Read/don't read starting vector of uniquenesses

Nonstandard/standard output

Calculates/does not calculate to more than 3-place accuracy by Ortega's method

Must be blank

Columns 45-49 are intended for experimentation and ordinarily would be left blank.

Nonstandard Parameter Cards (optional)

If Col. 46 in the parameter cards is 1, two nonstandard parameter cards must be inserted at this point. The user should refer to the write-up for the definition of these parameters.

Data-Format Card

The standard FORTRAN variable format is followed; i.e., a card of the form (nFw.d) must be punched.

Correlation Cards

The correlation cards punched according to the format specified in nonstandard parameter cards must be prepared in row order, each row beginning on a new card. Only the lower half of the correlation matrix excluding the main diagonal is punched; i.e., there will be \( P - 1 \) rows of input.

Starting Configuration (optional)

If Col. 47 in the parameter card is 1, an arbitrary starting configuration may be entered. This will consist of

(a) a format card

(b) the starting uniquenesses for the variables punched according to the format (a) above.

continued
Stop Card
At the end of the last case, a card punched STOP inCols. 1–4 must be placed.

The input deck should be in the following order.

Description of Output

The output is dependent upon the options selected inCols. 41–44 of the parameter card. For purposes of illustration the sample input did not contain any punches after Col. 40, so that a minimum output resulted. The following items are printed:

1. input parameters
2. maximum likelihood solution (labeled "unrotated factor matrix")
3. uniqueness for each variable
4. varimax solution

continued
(5) a test of goodness of fit for the specified number of factors.

Note: With items (2), (3), and (4) there appear a string of numbers for diagnostic purposes. These should be ignored.

REFERENCES

SAMPLE INPUT

A listing of the input data cards and the related output are presented below. Although the data used have the entire matrix punched, it would not have been necessary. Data punched through the diagonal value for each row would have been sufficient.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3</td>
<td>305</td>
<td>40</td>
<td>4 1.0</td>
</tr>
<tr>
<td>21</td>
<td>0.84600</td>
<td>1.00000</td>
<td>0.88100</td>
<td>0.80600</td>
</tr>
<tr>
<td>31</td>
<td>0.80500</td>
<td>0.88100</td>
<td>1.00000</td>
<td>0.80100</td>
</tr>
<tr>
<td>41</td>
<td>0.85900</td>
<td>0.82600</td>
<td>0.80100</td>
<td>1.00000</td>
</tr>
<tr>
<td>51</td>
<td>0.47300</td>
<td>0.37600</td>
<td>0.38000</td>
<td>0.45600</td>
</tr>
<tr>
<td>61</td>
<td>0.39800</td>
<td>0.32600</td>
<td>0.31900</td>
<td>0.32900</td>
</tr>
<tr>
<td>71</td>
<td>0.30100</td>
<td>0.27700</td>
<td>0.23700</td>
<td>0.32700</td>
</tr>
<tr>
<td>81</td>
<td>0.38200</td>
<td>0.41500</td>
<td>0.34500</td>
<td>0.36500</td>
</tr>
</tbody>
</table>

STOP
SAMPLE OUTPUT

UNRESTRICTED MAXIMUM LIKELIHOOD FACTOR ANALYSIS
EIGHT VARIABLES—ALL 41 ON BLANK
5/29/69
PARAMETERS

\[ \begin{align*}
    P &= 8 \\
    K1 &= 3 \\
    K2 &= 3 \\
    N &= 305 \\
    MAXITER &= 40 \\
    CRV &= 1.000
\end{align*} \]

MAXIMUM LIKELIHOOD SOLUTION FOR 3 FACTORS

UNROTATED FACTOR MATRIX

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>8</th>
<th>3</th>
<th>305</th>
<th>5</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.846</td>
<td>-0.189</td>
<td>0.348</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.000</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.881</td>
<td>-0.055</td>
<td>0.164</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.826</td>
<td>-0.160</td>
<td>0.368</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.376</td>
<td>-0.176</td>
<td>-0.031</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.326</td>
<td>-0.725</td>
<td>-0.093</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.277</td>
<td>-0.704</td>
<td>-0.130</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.415</td>
<td>-0.343</td>
<td>-0.204</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

continued
### Unique Variances

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>8</th>
<th>305</th>
<th>6</th>
<th>9</th>
<th>0.127</th>
<th>0.000</th>
<th>0.194</th>
<th>0.156</th>
<th>0.090</th>
<th>0.359</th>
<th>0.411</th>
<th>0.491</th>
</tr>
</thead>
</table>

### Varimax-Rotated Factor Matrix

|     | 0  | 8  | 3  | 305 | 5  | 9  | 0.883 | 0.275 | -0.129 | 0.936 | 0.204 | 0.287 | 0.872 | 0.196 | 0.088 | 0.875 | 0.239 | -0.145 | 0.234 | 0.319 | -0.105 | 0.185 | 0.779 | -0.021 | 0.129 | 0.737 | 0.004 | 0.254 | 0.648 | 0.157 |

### Test of Goodness of Fit for 3 Factors

CHISQUARE WITH 8 DEGREES OF FREEDOM IS 22.593
PROBABILITY LEVEL IS 0.00
COST ESTIMATE

For the job listed on the Sample Input, the total running time on the IBM 360/65 amounted to 3.80 seconds. At the current rate at Educational Testing Service (a rather complicated function of the ratio of CPU time to actual wall-clock running time), the total computer charge for the run amounted to $10.40. (The wall-clock/CPU-clock ratio was approximately 25:1.)

Charge to user = computer time + postage and handling + network overhead

= $10.40 + $5.00 + network overhead = $15.40

CONTENTS—UMLFA

pages

1. Identification & Abstract
2. User Instructions
3. I/O
4. Cost—Contents
Rao Constellation and Distance Analysis

DISCRIM2

Michigan State University
Computer Institute for Social Science
Research

Stuart Thomas
Computer Institute for Social Science
Research
Michigan State University

FORTRAN

CDC 3600

Decks and listings presently available

Dr. Anders Johanson, Programming Supervisor, Applications Programming, Computer Laboratory, Computer Center, Michigan State University, East Lansing, Mich. 48823
Tel.: (517) 355-4684

This program implements the method titled "Constellation and Distance Analysis" when first published by C.R. Rao¹ and later called "Multiple Discriminant Analysis" when presented independently by Bryan² and by Lubin. The method operates on a set of variates measured on individuals in several groups. It determines linear combinations of the variates, called discriminant functions, which maximize the ratio of between-group variability to pooled, within-group variability, producing the output listed below. The user can have the program handle data input or write his own subroutine to read data and perform preliminary data manipulations.

Output

1. Job Description
   User comments
   Number of groups and variables
   Options selected

continued
2. **Group Summary Statistics**
   - Identification
   - Number observations specified on group card
   - Mode of data input and input unit
   - Format statement (if data read by program rather than user)
   - First observation in the group (raw and transformed)
   - Variable means and variances
   - Variable intercorrelation matrix (optional)

3. **Overall Data Summary**
   - Overall means
   - Overall variances
   - Overall correlation matrix

4. **Discriminant Function Information**
   - Discriminant criterion
   - Percent trace accounted for by criterion
   - Rao's chi-square and degrees of freedom for the function
   - Function weights for raw data
   - Function weights for data adjusted to unit variances (optional)
   - Mean discriminant score for each group

5. **Overall Discriminant Statistics**
   - Group centroids in discriminant space (optional)
   - Intercentroid distance matrix (optional)
   - Back solution of discriminant equation (optional)

6. **Discriminant Scores (optional)**

**Capacity**
- Number of variables must not exceed 30
- Number of groups must not exceed 50
- Number of observations in any given group must not exceed 9,999
- Number of scores (if requested) must not exceed 10 per respondent

**REFERENCES**


USER INSTRUCTIONS

DISCRIM is called from the CISSR library tape by the name DISCRIM2. The routine is available in relocatable object deck form; however, unless extensive recoding of data is required, use of the library tape is recommended. The user may supply his own recoding subroutine for use with the object deck form.

Complete instructions on the options available and the use of the program are found in the references cited below.

REFERENCES


SAMPLE INPUT

`JOB, 015525, THOMAS, 3.0, APHL`  
`EQUIP, 27=(CISSR), MT(193), KO`  
`LOADMAIN, 27, 3.0, 3000`  

`$ROUTINE, DISCRIM2`  

`$DISCRIM TEST DATA`  

<table>
<thead>
<tr>
<th>SIZE</th>
<th>3</th>
<th>5</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCORES</td>
<td>2</td>
<td>PU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GROUP</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(5F5.0)

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>4</td>
<td>5</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>9</td>
<td>12</td>
<td>16</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>9</td>
<td>7</td>
<td>8</td>
<td>6</td>
</tr>
</tbody>
</table>

(5F5.0)

<table>
<thead>
<tr>
<th>11</th>
<th>13</th>
<th>15</th>
<th>17</th>
<th>19</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>14</td>
<td>15</td>
<td>15</td>
<td>19</td>
<td>21</td>
</tr>
<tr>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
</tr>
</tbody>
</table>

(5F5.0)

<table>
<thead>
<tr>
<th>21</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>43</td>
<td>49</td>
<td>65</td>
<td>71</td>
</tr>
<tr>
<td>23</td>
<td>45</td>
<td>48</td>
<td>64</td>
<td>72</td>
</tr>
<tr>
<td>24</td>
<td>47</td>
<td>47</td>
<td>66</td>
<td>73</td>
</tr>
</tbody>
</table>

ENDDATA
SAMPLE OUTPUT

MULTIPLE DISCRIMINANT ANALYSIS
IMPLEMENTED BY COMPUTER INSTITUTE FOR SOCIAL SCIENCE RESEARCH
MICHIGAN STATE UNIVERSITY

DISCRIM TEST DATA
SIZE CARD SPECIFIES 5 VARIABLES, 3 GROUPS
SCORES CARD REQUESTS 2 SCORES PER RESPONDENT OUTPUT TO 00

DATA SUMMARY FOR GROUP 1
SIZE CARD SPECIFIES 5 OBSERVATIONS
FORMAT FOR GROUP 1 = (95,6)
DATA INPUT BY PROGRAM FROM LOGICAL UNIT 60
FIRST RAW OBSERVATION FOR GROUP 1
VARIABLE 1 2 3 4 5
VALUE 1,0000=000 2,0000=000 3,0000=000 4,0000=000 5,0000=000
FIRST TRANSFORMED OBSERVATION FOR GROUP 1
VARIABLE 1 2 3 4 5
VALUE 1,0000=000 2,0000=000 3,0000=000 4,0000=000 5,0000=000
COMPUTATIONS BASED ON 5 OBSERVATIONS AFTER DROPPING 0
VARIABLE MEANS AND VARIANCES IN GROUP 1
VARIABLE 1 2 3 4 5
MEAN 3,0000=000 5,0000=000 6,0000=000 8,0000=000 7,4000=000
VARIANCE 2,5000=000 6,7000=000 5,2000=000 8,5000=000 3,1300=001

CORRELATION MATRIX FOR GROUP 1

1 2 3 4 5
1 1,0000=000 9,1627=001 6,9338=001 6,4233=001 8,4957=001
2 9,1627=001 1,0000=000 7,1196=001 9,6517=001 1,4847=001
3 6,9338=001 7,1196=001 1,0000=000 9,7769=001 4,6246=001
4 5,4233=001 9,6517=001 9,7769=001 1,0000=000 5,5177=001
5 8,4957=001 1,4847=001 4,6246=001 9,5177=001 1,0000=000

continued
DATA SUMMARY FOR GROUP  2
SIZE CARD SPECIFIES  5 OBSERVATIONS
DATA INPUT BY PROGRAM FROM LOGICAL UNIT 60
FIRST RAW OBSERVATION FOR GROUP  2

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALUE</td>
<td>1,1000+001</td>
<td>1,3000+001</td>
<td>1,9000+001</td>
<td>1,7000+001</td>
<td>1,9000+001</td>
</tr>
</tbody>
</table>

FIRST TRANSFORMED OBSERVATION FOR GROUP  2

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALUE</td>
<td>1,1000+001</td>
<td>1,3000+001</td>
<td>1,9000+001</td>
<td>1,7000+001</td>
<td>1,9000+001</td>
</tr>
</tbody>
</table>

COMPUTATIONS BASED ON  5 OBSERVATIONS AFTER DROPPING  0

VARIABLE MEANS AND VARIANCES IN GROUP  2

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEAN</td>
<td>1,0600400</td>
<td>1,4200401</td>
<td>1,5000.001</td>
<td>1,6600+001</td>
<td>1,0400.001</td>
</tr>
<tr>
<td>VARIANCE</td>
<td>2,5000.000</td>
<td>1,7000+001</td>
<td>2,1000+001</td>
<td>5,3000+000</td>
<td>7,0000+000</td>
</tr>
</tbody>
</table>

CORRELATION MATRIX FOR GROUP  2

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.0000000</td>
<td>0.4887001</td>
<td>4.4721-001</td>
<td>3.4340-001</td>
<td>3.8651+001</td>
</tr>
<tr>
<td>2</td>
<td>0.4887+001</td>
<td>1.0000+000</td>
<td>9.340+001</td>
<td>6.996+001</td>
<td>2.8123+001</td>
</tr>
<tr>
<td>3</td>
<td>4.4721-001</td>
<td>9.340+001</td>
<td>1.0000+000</td>
<td>7.6787+001</td>
<td>2.4693+001</td>
</tr>
<tr>
<td>4</td>
<td>3.4340+001</td>
<td>6.996+001</td>
<td>7.6787+001</td>
<td>1.0000+000</td>
<td>4.7782+001</td>
</tr>
<tr>
<td>5</td>
<td>3.8651+001</td>
<td>2.8123+001</td>
<td>2.4693+001</td>
<td>4.7782+001</td>
<td>1.0000+000</td>
</tr>
</tbody>
</table>

DATA SUMMARY FOR GROUP  3
SIZE CARD SPECIFIES  4 OBSERVATIONS
DATA INPUT BY PROGRAM FROM LOGICAL UNIT 60
FIRST RAW OBSERVATION FOR GROUP  3

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALUE</td>
<td>2,1000+001</td>
<td>4,0000+001</td>
<td>5,0000+001</td>
<td>6,0000+001</td>
<td>7,0000+001</td>
</tr>
</tbody>
</table>

FIRST TRANSFORMED OBSERVATION FOR GROUP  3

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALUE</td>
<td>2,1000+001</td>
<td>4,0000+001</td>
<td>5,0000+001</td>
<td>6,0000+001</td>
<td>7,0000+001</td>
</tr>
</tbody>
</table>

COMPUTATIONS BASED ON  4 OBSERVATIONS AFTER DROPPING  0

continued
### Variable Means and Variances in Group 3

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEAN</td>
<td>2,2500+000</td>
<td>4,3750+000</td>
<td>4,8500+000</td>
<td>6,3750+000</td>
<td>7,1500+000</td>
</tr>
<tr>
<td>VARIANCE</td>
<td>1,6667+000</td>
<td>8,9167+000</td>
<td>1,6667+000</td>
<td>6,9167+000</td>
<td>1,6667+000</td>
</tr>
</tbody>
</table>

### Correlation Matrix for Group 3

```
  1  2  3  4  5
1 1.0000+000 9.9438+001 9.9438+001 9.9438+001 9.9438+001
2 9.9438+001 1.0000+000 9.9438+001 9.9438+001 9.9438+001
3 -1.0000+000 9.9438+001 1.0000+000 9.9438+001 -1.0000+000
4 9.9438+001 9.9438+001 -8.3456+001 1.0000+000 9.9438+001
5 1.0000+000 9.9438+001 1.0000+000 9.9438+001 1.0000+000
```

### Overall Means and Variances Based on 14 Total Observations

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEAN</td>
<td>1,2143+001</td>
<td>1,9429+001</td>
<td>2,1429+001</td>
<td>2,1429+001</td>
<td>2,1429+001</td>
</tr>
<tr>
<td>VARIANCE</td>
<td>6,7363+000</td>
<td>2,7503+000</td>
<td>3,13819+002</td>
<td>2,0070+002</td>
<td>8,4926+002</td>
</tr>
</tbody>
</table>

```
  1  2  3  4  5
1 1.0000+000 9.4330+000 9.2378+001 9.5433+000 8.7013+000
2 9.4330+000 1.0000+000 9.9012+001 9.9167+001 9.6863+001
3 9.2378+001 9.9012+001 1.0000+000 9.9426+001 9.7563+001
4 9.5433+000 9.9167+001 9.9426+001 1.0000+000 9.8393+001
5 8.7013+000 9.6863+000 9.7563+000 9.8393+001 1.0000+000
```
DISCRIMINANT FUNCTION NUMBER 1
DISCRIMINANT CRITERION FOR THIS FUNCTION 1.7242E+002
PERCENT OF TRACE 9.2960E+001
RAO'S CHI-SQUARE FOR THIS FUNCTION IS 5.1597E+001 WITH 6 DEGREES OF FREEDOM
NORMED FUNCTION WEIGHTS

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>WEIGHT</td>
<td>9.9922E-001</td>
<td>5.2612E-001</td>
<td>6.0288E-001</td>
<td>1.1874E+002</td>
<td>2.3631E+002</td>
</tr>
</tbody>
</table>

STANDARDIZED FUNCTION WEIGHTS

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>STD. WT.</td>
<td>3.3011E-001</td>
<td>5.8571E-001</td>
<td>7.3655E-001</td>
<td>1.9999E+002</td>
<td>4.6227E+002</td>
</tr>
</tbody>
</table>

GROUP MEAN SCORES ON THIS FUNCTION

<table>
<thead>
<tr>
<th>GROUP</th>
<th>MEAN SCR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>6.4457E+000</td>
</tr>
</tbody>
</table>

DISCRIMINANT FUNCTION NUMBER 2
DISCRIMINANT CRITERION FOR THIS FUNCTION 1.3057E+001
PERCENT OF TRACE 7.0398E+000
RAO'S CHI-SQUARE FOR THIS FUNCTION IS 2.6431E+001 WITH 4 DEGREES OF FREEDOM
NORMED FUNCTION WEIGHTS

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>WEIGHT</td>
<td>9.0159E-001</td>
<td>4.3111E-001</td>
<td>2.5524E+002</td>
<td>2.0384E+002</td>
<td>1.4650E+002</td>
</tr>
</tbody>
</table>

STANDARDIZED FUNCTION WEIGHTS

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
</table>

GROUP MEAN SCORES ON THIS FUNCTION

<table>
<thead>
<tr>
<th>GROUP</th>
<th>MEAN SCR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>6.6098E+001</td>
</tr>
</tbody>
</table>

continued
GROUP CENTROIDS IN DISCRIMINANT SPACE

1 2 3
1 -4.9457e+000 -9.7613e+000 -4.1220e+001
2 6.6098e+001 0.0796e+000 2.7706e+000

INTERCENTROID DISTANCES IN DISCRIMINANT SPACE

1 2 3
1 0.0000e+000 6.9274e+000 3.6336e+001
2 6.9274e+000 0.0000e+000 3.2130e+001
3 3.6336e+001 3.2130e+001 0.0000e+000

0 OBSERVATIONS DROPPED FROM SCORES COMPUTATION FOR GROUP 1 DUE TO PARITY ERRORS
0 OBSERVATIONS DROPPED FROM SCORES COMPUTATION FOR GROUP 2 DUE TO PARITY ERRORS
0 OBSERVATIONS DROPPED FROM SCORES COMPUTATION FOR GROUP 3 DUE TO PARITY ERRORS

BACK SOLUTION OF DISCRIMINANT EQUATION

1 2
1 -1.7462e+007 1.9707e+008
2 5.4295e+008 3.1469e+008
3 2.2631e+007 9.0780e+008
4 -1.8294e+007 1.3411e+007
5 9.2387e+007 1.2293e+007
COST ESTIMATE

For the job listed on the Sample Output, the total processing time was 17.007 seconds. At the current rate for the Michigan State University's CDC 3600 ($245/hr.), the computer time cost $1.16 plus a charge for input and output. A 10% surcharge will be assessed if billed to a non-University account. In addition, there is a consulting fee of $10/hr. for work done by the Applications Programming Group.

Charge to user = computer costs + consulting + network overhead
              = $1.20 (approx.) + consulting + network overhead

CONTENTS—DISCRIM2

pages  Identification & Abstract
       1-2  User Instructions
       3    I/O
       5-10  Cost—Contents
FUNCTIONAL ABSTRACT

The latest version of the multidimensional scaling program written by J.B. Kruskal of Bell Telephone Labs is available. In addition to the improvements made from earlier versions, a much faster sort has been incorporated into the program and a multi-calculation facility using random starting configurations within one machine run, has been made available. This feature, in addition to being more efficient than the old technique of punching cards for subsequent runs, increases the chances of converging to a global minimum within one machine run with the corresponding savings in time and cost.

A modification to the output was made so that normally only the best of many possible configurations is printed. However, each final configuration and even the individual iterations may be printed (as was done in Kruskal's version) if desired.

The program will handle 60 subjects scaled in up to ten dimensions. Calculations can be repeated up to 99 times on as many as 1800 data values.

continued
REFERENCES


USER INSTRUCTIONS

Input

Program input may be broken down into five types of cards.

1. Control Cards governing the types of calculations desired
2. a Title Card identifying the run
3. a Parameter Card indicating size of matrix, replications, perturbations and print options
4. a Format Card identifying the form of the input data
5. the data

Control Cards

These are used, basically, to override preset program constants, if desired. They may be inserted in any order with the restriction that the "similarities" or "dissimilarities" option must be the last specified. They may be punched one per card or connected by commas. For a complete description see Ref. 1. The following may be used. (Punch # for = on an 029 keypunch.)

Control Options
(begin in Col. 1) Explanation
DIMMAX = integer (pre-set value is in parentheses)
DIMMIN = integer Maximum dimension (2)
DIMDIF = integer Minimum dimension (2)
ITERATIONS = integer Step size from maximum to minimum (1)
DIAGONAL PRESENT Limitations for convergence (50)
DIAGONAL ABSENT Data specification (present)
MATRIX Data specification (matrix)
HALFMATRIX VALUES < cutoff are discarded (0.0)
CUTOFF = floating pt. Indicates weights for data follow (no weights)
WEIGHTS Metric space (2)
R = floating pt. > 1 Equal distance stress contribution (PRIMARY)
PRIMARY continued
Control Options
(begin in Col. 1)
SFORM 1
SFORM 2

Explanations
(pre-set value is in parentheses)
Different stress formulae (see Ref. 1) [SFORM 1]
Note: Ref. 1 indicates that the standard option has proven superior in all cases tested.

NOCARDS
CARDS

Final configuration punch option (NOCARDS)
Note: When CARDS is requested, only the best configuration is punched.

CONFIGURATION
Identification of configuration deck (no configuration)
STRMIN = floating point
Satisfactory stress value (.01)
SFGRMN = floating point
Satisfactory gradient minimum (0.0)
SRATST = floating point
Satisfactory change in successive stress values (.999)
SIMILARITIES
DISSIMILARITIES
Definition of data measurement (dissimilarities)
Note: This option must be the last one specified.

Title Card
The Title Card may have any identification punched in Col. 1–72. It follows the SIMILARITIES or DISSIMILARITIES Card.

Parameter Card
The Parameter Card follows the Title Card.

Columns | Contents
---|---
1–3 | Size of data matrix
4–6 | No. of replications (i.e., No. of values for each matrix entry)
7–9 | No. of perturbations (i.e., No. of separate calculations with initial random configuration, ≤ 99)
10–12 | 1: print iteration values
      | 0: do not print iteration values

continued
Columns | Contents
---|---
13-15 | 1: print final configuration for each perturbation 0: do not print final configuration

Format Card for Input Data

This is a regular Fortran format enclosed by parentheses,

Ex: (10 F 8.3)

A slight modification of this format is used to print the data as part of the output; care must be taken to assure that the entire number is printed. Since decimals on the Data Cards will override the format specifications, only the field width need be of concern when punching the Data Cards. However, data punched .821, .342, etc. with a format of (8 F 5.0) will read correctly but will print as zeros. With a format of F w.d, w should be at least three greater than the number of significant digits desired.

Data Cards

Data matrix input is by rows according to format specified on the Format Card. Begin each row on a new card, the count will be determined by the size of the data matrix. If a half matrix is being read, it is convenient to consider it as the lower triangular matrix, including or excluding the diagonal.

Following the last Data Card, a Control Card, COMPUTE, is placed. A Stop Control Card follows the last Compute Card for the entire data deck. Multiple cases may be handled on one machine run.

Compute Card

Columns | Contents
---|---
1-7 | COMPUTE

Stop Card

Columns | Contents
---|---
1-4 | STOP

System Control Cards

JOB Card will be prepared by ETS personnel.

continued
System Card 1
Columns Contents
1-37 //STEPNAME EXEC GITNGO,NAME=KRUSKAL

System Card 2
Columns Contents
1-19 //GO.SYSIN DD *

System Card 3
Columns Contents
1-2 //

Job Deck
Job Card
System Card 1
System Card 2
Data Deck
System Card 3

Output
The output is virtually self-explanatory. An added feature is
the printout of the various control cards and the input data
matrix so that visual checking for errors may be made.

Normally, the stress and best configuration are printed with
a history of the stresses, number of iterations, and a conver-
gence code for each perturbation. From rather limited experi-
ence it has been found that a minimum is generally located
within ten perturbations (i.e., no smaller values were found
when 40 or 50 were used).

When the CARDS punch option is requested, only the best con-
figuration is punched.

Limitations
The present program will handle 60 objects scaled in up to ten
dimensions. 1800 data values may be read, and up to 99 per-
turbations may be performed.

REFERENCES
1. Kruskal, J.B., "Nonmetric Multidimensional Scaling: A
SAMPLE INPUT

//KRT2  JOB (....
//STEPNAME EXEC GITNGO,NAME=KRUSKAL
//GO.SYSIN DD *
DIMMIN=1
ITERATIONS=50
SIMILARITIES
TEST CASE WITH NEW WRITEUP
  5 1 10
(5F5.2)
9.3 0.57 6.1 2.0 1.5
1.0 8.5 5.5 3.3 4.0
2.1 5.3 7.3 3.9 1.1
5.0 6.2 1.6 8.9 2.8
1.3 4.1 1.0 2.5 8.8
COMPUTE
STOP
/*
//

10/70
SAMPLE OUTPUT

CONTROL CARDS
DIMN=2
ITERRATIONS=5
SIMILARITIES
TEST CASE WITH NEW WRITEUP
NO. OBJECTS=5 REPLICATIONS=10 ITERATION PRINT=0 PERTURBATION PRINT=0

THE BEST CONFIGURATION OF 5 POINTS IN 2 DIMENSIONS
HAS STRESS 0.037

BEST CONFIGURATION
1 2
1 -1.052 -0.677
2 0.529 0.769
3 -0.750 0.538
4 0.059 -0.350
5 1.215 -0.300

HISTORY OF PERTURBATIONS
CODE STRESS ITERATIONS
2.0 0.225 20.0
2.0 0.230 20.0
2.0 0.184 23.0
2.0 0.037 24.0
2.0 0.037 26.0
2.0 0.037 26.0
2.0 0.037 32.0
2.0 0.037 30.0
2.0 0.037 34.0
2.0 0.037 30.0

CODES: -1<ZERO STRESS, 12<MINIMUM ACHIEVED,
13<SATISFACTORY STRESS REACHED, 14<MAXIMUM ITERATIONS USED

THE BEST CONFIGURATION OF 5 POINTS IN 1 DIMENSIONS
HAS STRESS 0.317

BEST CONFIGURATION
1
1 -1.493
2 1.451
3 -0.042
4 -0.533
5 0.617

HISTORY OF PERTURBATIONS
CODE STRESS ITERATIONS
2.0 0.409 17.0
2.0 0.413 25.0
2.0 0.391 22.0
2.0 0.318 20.0
2.0 0.317 27.0
2.0 0.447 15.0
2.0 0.399 15.0
2.0 0.378 17.0
2.0 0.423 16.0
2.0 0.367 20.0

CODES: -1<ZERO STRESS, 12<MINIMUM ACHIEVED,
13<SATISFACTORY STRESS REACHED, 14<MAXIMUM ITERATIONS USED

***************************************************************************************
COST ESTIMATE

The pricing algorithm for the Educational Testing Service is based on a complex formula involving the amount of core usage, card-processing, tape and dedicated device usage, lines printed and central processor unit (CPU) elapsed time. Computer costs for the job included in the Sample Input were $3.66.

\[
\text{Charge to user} = \text{computer costs} + \text{postage and handling} + \text{network overhead} \\
= $3.66 + $5.00 + \text{network overhead} \\
= $8.66 + \text{network overhead}
\]

CONTENTS—KRUSKAL

<table>
<thead>
<tr>
<th>pages</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>Identification &amp; Abstract</td>
</tr>
<tr>
<td>3-6</td>
<td>User Instructions</td>
</tr>
<tr>
<td>7-8</td>
<td>I/O</td>
</tr>
<tr>
<td>9</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>
FUNCTIONAL ABSTRACT
A flexible plot routine originally written for the 7090 at the University of Michigan and adapted to their system, has been modified for the operating system at ETS. Although the program is written in assembly language, the necessary linkage to FORTRAN and FORTRAN I/O has been written. It is intended to be called, therefore, by a FORTRAN program and will produce plots in the normal output stream as determined by the various calls, to different entry points in the subroutine. The standard approach is to call PLOT1, PLOT2, PLOT3, and PLOT4 in that order. PLOT1 sets up the information required to construct the graph, PLOT2 prepares the grid and sets up the information required by PLOT3 to place points in the graph, PLOT3 places the plotting characters at the specified points in the graph, and PLOT4 prints the completed graph with values along the X and Y axes and a centered vertical label down the left side of the graph.
USER INSTRUCTIONS

PLOT1

PLOT1 sets up the information required to construct the graph. It consists of five arguments, the first of which is the first location of a five-word vector. The calling sequence is

CALL PLOT1 (NSCALE(1), NHL, NSBH, NVL, NSBV)

The five arguments are defined as follows.

NSCALE(1)—controls the scaling and number of decimals printed. If this is 0, the values 0, 3, 0, 3 are used for NSCALE(2) through NSCALE(5), respectively.

NSCALE(2)—if this has the value n, the numbers printed along the y axis are $10^n$ times their true value.

NSCALE(3)—the number of decimal places printed for y values.

NSCALE(4)—if this has the value n, the numbers printed along the x axis are $10^n$ times their true value.

NSCALE(5)—the number of decimal places printed for X values.

NHL—number of horizontal lines—must be two or greater.

NSBH—number of spaces between horizontal lines—must be one or greater.

NVL—number of vertical lines—must be two or greater.

NSBV—number of spaces between vertical lines—must be one or greater.

NOTE: If different values for NSCALE(1) through NSCALE(5) are desired, then the call to NSCALE(1) should be the name of a five-entry array containing the desired values.

PLOT2

PLOT2 prepares the grid and sets up the information required by PLOT3 to place a point correctly in the graph. The calling sequence is

CALL PLOT2 (IMAGE, XMAX, XMIN, YMAX, YMIN)

If PLOT1 has not been called prior to PLOT2, the following default configuration is used: NSCALE(1) = 0, NHL = 6, NSBH = 9, NVL = 11, NSBV = 9. The five arguments are defined as follows.

IMAGE—the region used to hold the image of the graph. A dimension of 2000 should be sufficient.

continued
XMAX—largest X value to be plotted.
XMIN—smallest X value to be plotted.
YMAX—largest Y value to be plotted.
YMIN—smallest Y value to be plotted.

**PLOT3**
PLOT3 places the plotting character in the graph for each point (X,Y). Its calling sequence is

```
CALL PLOT3 (BCD, X, Y, NDATA, INT)
```

The five arguments are defined as follows.

BCD—plotting character used. Example: (1H*).
X—the first element of a floating point vector of X values.
Y—the first element of a floating point vector of Y values.
NDATA—the number of points to be plotted.
INT—the number of bytes between the characteristics of numbers used as coordinates (short = 4, long = 8).

**PLOT4**
PLOT4 will print the completed graph with values along the X and Y axes with a centered vertical label down the left side. The calling sequence is

```
CALL PLOT4 (NCHAR, LABEL)
```

The two arguments are defined as follows.

NCHAR—the number of characters in the label.
LABEL—the location of the first character of the label. The following example is representative.

```
CALL PLOT4 (18, 'ANY VERTICAL LABEL')
```

**System Control Cards**

JOB Card will be supplied by ETS.

System Card 1

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-22</td>
<td>//PROG EXEC FORTGCLS,</td>
</tr>
</tbody>
</table>

continued
System Card 2
Columns Contents
1-32 // REGION.GO=120K,ACCT.GO=0200

System Card 3
Columns Contents
1-18 //FORT.SYSIN DD *

System Card 4
Columns Contents
1-36 //LKED.LIB DD DSN=OCS.SAPGM,DISP=SHR

System Card 5
Columns Contents
1-19 //LKED.SYSIN DD *

System Card 6
Columns Contents
11-27 INCLUDE LIB(PLOT)

System Card 7
Columns Contents
1- 2 /*

System Card 8
Columns Contents
1- 2 //

Job Deck
JOB Card
System Card 1
System Card 2
System Card 3
Input Deck
System Card 4
System Card 5
System Card 6
System Card 7
System Card 8

continued
The preceding information should suffice for a high percentage of the graphs required. A subroutine has been written to insert BCD information in the body of the graph, and additional features exist to handle semi-log and log-log plots, but have not been completely checked out as yet. Anyone wishing to use these options should consult the contact person.
EXAMPLE 1

//(.JOB CARD)
//PROG EXEC FORTGCLG,
//REGION.GD=120K,ACCT.GD=02000
//FORT.SYSIN DD *
DIMA(100),X2(100),X3(100).GRAPH(2000),IN(5)
REAL*8 GRAPH2(1000)
COMMON GRAPH,NVL,NSBV
EQUIVALENCE(GRAPH,GRAPH2)
DIMA A(100)
NGT = 100
DO Z I=1,100
A(I) = I
X1(I) = A(I)/100.
X2(I) = 1.-X1(I)
X3(I) = X1(I)*X1(I)
NHL=6
NSBH=9
NVL=6
NSBV=19
IN(1) = 1
IN(2) = 0
IN(3) = 1
IN(4) = 0
IN(5) = 1
CALL PLOT1(IN,NHL,NSBH,NVL,NSBV)
CALL PLOT2(GRAPH,100.,0.,1.,0.)
CALL PLOT3(A,X1,NGT,4)
CALL PLOT3(A,X2,NGT,4)
CALL PLOT3(A,X3,NGT,4)
WRITE (6,5)
5 FORMAT(1H1,40X,'SO YOU THINK YOU HAVE TROUBLE?-----')
CALL PLOT4('READE THIS VER/Y CAREFULLY')
WRITE (6,8)
8 FORMAT(1H0,40X,'WILL THESE SOON BE REVERSED?','SOX,'DBK')
END
//LKED.LIB DD DSN=OCS.SAPGM,DISP=SHR
//LKED.SYSIN DD *
INCLUDE LIB(INSH)
/*
INCLUDE LIB(PLOT)
// continued
EXAMPLE 2

Control cards are not listed in this example.

```fortran
REAL*8 GRAPH2(1000)
EQUIVALENCE (GRAPH,GRAPH2)
DATA BLANK/' ',BCD/'**'/
POL(ANGLE) = .25 * ANGLE
RMAX = 4.
NSBV=79
NYL=2
CALL PLOT1 (9,2,47,2,79)
CALL PLOT2 (GRAPH,RMAX,*RMAX,RMAX,*RMAX)
DO 1 I=1,20:0
 1 GRAPH(I)=BLANK
  DELTAY = RMAX/24.
  DELTA*X=RMAX/40.
   IMAX = 2.*RMAX+1.
  DO 4 I=1,49
 4 CALL PLOT3(1HI,0.,RMAX-FLOAT(I-1)*DELTAY,1,4)
  DO 5 I=1,IMAX
 5 CALL PLOT3(1H+,0.,RMAX-FLOAT(I-1),1,4)
  DO 6 I=1,49
 6 CALL PLOT3(1H-,RMAX-FLOAT(I-1)*DELTAY,0.,1,4)
  DO 7 I=1,IMAX
 7 CALL PLOT3(1H+,-RMAX+FLOAT(I-1),0.,1,4)
  DO 12 I=1,140
  12 THET = .1 * FLOAT(I)
   SWITCH = PLOT3(BCD,POL(THET)*COS(THET),POL(THET)*SIN(THET),1,4)
  CONTINUE
  CALL OMIT(3)
  CALL INSH(44,1,3,'90 DEGREES ')
  CALL INSH(2,26,3,'180 DEGREES ')
  CALL INSH(71,26,3,'0 DEGREES ')
  CALL INSH (44,47,3,'270 DEGREES ')
  WRITE (6,9)
  9 FORMAT(1H1,40X,' A SLIGHTLY DIFFERENT EXAMPLE')
  CALL PLOT4(26,'POLAR PLOT OF THE SPIRAL '
END
```
SAMPLE OUTPUT

EXAMPLE 1

SO YOU THINK YOU HAVE TROUBLES

MACHINE AVAILABILITY

PROGRAMMING DEMANDS

BLOOD PRESSURE

WILL THESE SOON BE REVERSIBLE?

continued
EXAMPLE 2

A SLIGHTLY DIFFERENT EXAMPLE

90 DEGREES

180 DEGREES

0 DEGREES

270 DEGREES

POLAR PLOT OF THE SPIRAL
COST ESTIMATE

The pricing algorithm for the Educational Testing Service is based on a complex formula involving the amount of core usage, card-processing, tape and dedicated device usage, lines printed and central processor unit (CPU) elapsed time. Computer costs for Example 1 in the Sample Input were $2.01.

Example 1
Charge to user = computer costs + postage and handling + network overhead
= $2.01 + $5.00 + network overhead
= $7.01 + network overhead

Computer costs for Example 2 in the Sample Input were $2.09.

Example 2
= $2.09 + $5.00 + network overhead
= $7.09 + network overhead

CONTENTS—PLOT

<table>
<thead>
<tr>
<th>pages</th>
<th>Identification &amp; Abstract</th>
<th>User Instructions</th>
<th>I/O</th>
<th>Cost—Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Identification &amp; Abstract</td>
<td>User Instructions</td>
<td>I/O</td>
<td>Cost—Contents</td>
</tr>
<tr>
<td>3–6</td>
<td>User Instructions</td>
<td>I/O</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7–10</td>
<td>I/O</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Cost—Contents</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
MSA-I is a program to map types (individuals having the same profile over a set of variables or items) onto an Euclidean space with minimum dimensionality. No assumptions are required about the underlying distributions, the scaling properties of the items, or their ordering. The only requirement is that the categories of each item be mutually exclusive and exhaustive. Types are represented as points in space, each item is a partition of the space, and each category is a region. All types who fall in a given category of a particular item are constrained to be closer to boundary markers of the same category than to delimiters of other categories of the same item. The program we have is a modification of the MSA-I package as distributed continued
by Dr. Lingoes. An option or modification provides additional plots for each category with the number inserted in the plots corresponding to the response to the item rather than the ID number as in the distributed version.

References and some copies of the substantial literature related to this problem are available through the contact person.

REFERENCE

### USER INSTRUCTIONS

#### Input Deck

**Title Card**

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-80</td>
<td>Title information</td>
</tr>
</tbody>
</table>

**Control Card (zeros and blanks treated alike)**

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>Number of variables ($\leq 50$)</td>
</tr>
<tr>
<td>5-8</td>
<td>Number of subjects ($\leq 100$)</td>
</tr>
<tr>
<td>9-12</td>
<td>Precoded switch</td>
</tr>
<tr>
<td></td>
<td>1: values are integers in the range 1-20</td>
</tr>
<tr>
<td></td>
<td>0 or blank: values are not in this range</td>
</tr>
<tr>
<td>13-16</td>
<td>If precoded switch is 1, leave this blank</td>
</tr>
<tr>
<td></td>
<td>If precoded switch is 0 or blank, punch number of</td>
</tr>
<tr>
<td></td>
<td>intervals into which data are to be coded.</td>
</tr>
<tr>
<td>17-20</td>
<td>Smallest frequency to be permitted after coding.</td>
</tr>
<tr>
<td></td>
<td>Blank indicates no lower limits are desired.</td>
</tr>
<tr>
<td>21-24</td>
<td>1: card output of coded data is desired</td>
</tr>
<tr>
<td></td>
<td>0: card output is not desired</td>
</tr>
<tr>
<td>25-28</td>
<td>Number of types of nonredundant profiles.</td>
</tr>
<tr>
<td></td>
<td>If program is to determine redundancies, set this</td>
</tr>
<tr>
<td></td>
<td>item to the number of subjects.</td>
</tr>
<tr>
<td>29-32</td>
<td>Minimum number of dimensions (default is 1)</td>
</tr>
<tr>
<td>33-36</td>
<td>Maximum number of dimensions (either 2 or 3)</td>
</tr>
<tr>
<td>37-40</td>
<td>Number of iterations (default is 25)</td>
</tr>
<tr>
<td>41-44</td>
<td>1: item plots for each item</td>
</tr>
<tr>
<td></td>
<td>0: no item plots</td>
</tr>
<tr>
<td></td>
<td>It is suggested that this be set to 1.</td>
</tr>
<tr>
<td>45-48</td>
<td>Cutoff criterion for coefficient of contiguity.</td>
</tr>
<tr>
<td></td>
<td>Floating point ($0 &lt; c \leq 1.00$)</td>
</tr>
<tr>
<td>49-52</td>
<td>Code to indicate mean is to be substituted for</td>
</tr>
<tr>
<td></td>
<td>missing data.</td>
</tr>
</tbody>
</table>

*continued*
Frequency Card Deck  (Use if Number of types of nonredundant profiles ≠ 0.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>Frequency for first type</td>
</tr>
<tr>
<td>3-4</td>
<td>Frequency for second type</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>71-72</td>
<td>Frequency for thirty-sixth type</td>
</tr>
</tbody>
</table>

If there are more than 36 types, continue onto another card.

Format Card

Data Deck
Multiple cases may be run, each with its own Input Deck.

Job Deck
The program is currently available and may be executed as follows.

//JOB card (will be provided by ETS personnel)
//EXEC GITNGO,NAME=MSAX
//GO.SYSIN DD *
(Insert Input Deck or Decks here)

Output
The output will depend upon the options selected in the Control Card.
SAMPLE INPUT

//DKMSA  JOB (....
//EXEC GITNGO, NAME=MSAX
//GO SYSIN DD *
I TEST FOR G-L(MSA-I) FOR A PERFECT UNIDIMENSIONAL SCALE OF 3 CATAGORIES.

3 7 3
7 2 2 1 1 0
(3F1.0)
333
332
322
222
221
211
111
*/

8/70
SAMPLE OUTPUT

TEST FOR G-L(MsA-I) FOR A PERFECT UNIDIMENSIONAL SCALE OF 3 CATEGORIES.

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>SCORE RANGE</th>
<th>CODED INTERVAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.100000E 01 TO 0.300000E 01</td>
<td>0.100000E 01</td>
</tr>
<tr>
<td>2</td>
<td>0.100000E 01 TO 0.300000E 01</td>
<td>0.100000E 01</td>
</tr>
<tr>
<td>3</td>
<td>0.100000E 01 TO 0.300000E 01</td>
<td>0.100000E 01</td>
</tr>
</tbody>
</table>

FREQUENCY DISTRIBUTION FOR RANKED CODED DATA

<table>
<thead>
<tr>
<th>VAR.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SCORES FOR EACH SUBJECT ON EACH VECTOR

<table>
<thead>
<tr>
<th>VECTOR</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUBJECT</td>
<td>1</td>
<td>-529. 509.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>-409. -69.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>-230. -320.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0. 0.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>230. 320.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>409. 69.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>529. 509.</td>
<td></td>
</tr>
</tbody>
</table>

LINGOES MULTIVARIATE ANALYSIS OF CONTINGENCIES

CATEGORY WEIGHTS

<table>
<thead>
<tr>
<th>VECTOR</th>
<th>ETA = .811</th>
</tr>
</thead>
<tbody>
<tr>
<td>CATEGORY</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>529. 213. -389.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>469. 0. -469.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>389. -213. -529.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

VECTOR 2 ETA = .368

| CATEGORY | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>509. -256. -96.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>289. -385. 289.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>86. -256. 509.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

continued
GUTTMAN-LINGOES OUTER-POINT SCALOGRAM ANALYSIS COORDINATES
FOR M = 2.

DIMENSION 1 2

-----------------------------------------------

<table>
<thead>
<tr>
<th>TYPE</th>
<th>1</th>
<th>-100.000</th>
<th>30.643</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>-77.180</td>
<td>-25.472</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-43.381</td>
<td>30.644</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>43.382</td>
<td>-74.921</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>77.180</td>
<td>-25.471</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>100.000</td>
<td>30.644</td>
<td></td>
</tr>
</tbody>
</table>

COEFFICIENT OF CONTIGUITY = 0.100000E 01 FOR 1 ITERATIONS.

OUTER-POINT MATRIX

-----------------------------------------------

| TYPE = 1 (4) | 3 3 -3 |
| TYPE = 2 (1) | 3 -3 -2 |
| TYPE = 3 (1) | -3 -2 2 |
| TYPE = 4 (2) | -2 2 -2 |
| TYPE = 5 (1) | 2 -2 -1 |
| TYPE = 6 (1) | -2 -1 1 |
| TYPE = 7 (1) | -1 1 1 |

continued
VECTOR PLOTS
VECTOR 2 PLOTTED AGAINST VECTOR 1

<table>
<thead>
<tr>
<th>VECTOR 1</th>
<th>VECTOR 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>96</td>
<td>96</td>
</tr>
<tr>
<td>92</td>
<td>92</td>
</tr>
<tr>
<td>88</td>
<td>88</td>
</tr>
<tr>
<td>84</td>
<td>84</td>
</tr>
<tr>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>76</td>
<td>76</td>
</tr>
<tr>
<td>72</td>
<td>72</td>
</tr>
<tr>
<td>68</td>
<td>68</td>
</tr>
<tr>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>60</td>
<td>60</td>
</tr>
<tr>
<td>56</td>
<td>56</td>
</tr>
<tr>
<td>52</td>
<td>52</td>
</tr>
<tr>
<td>48</td>
<td>48</td>
</tr>
<tr>
<td>44</td>
<td>44</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>28</td>
<td>28</td>
</tr>
<tr>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

VECTOR 2

```
-100  -90  -80  -70  -60  -50  -40  -30  -20  -10   0   10   20   30   40   50   60   70   80   90   100
```

```
-100  -90  -80  -70  -60  -50  -40  -30  -20  -10   0   10   20   30   40   50   60   70   80   90   100
```
COST ESTIMATE

The pricing algorithm for the Educational Testing Service is based on a complex formula involving the amount of core usage, card-processing, tape and dedicated device usage, lines printed and central processor unit (CPU) elapsed time. Computer costs for the job included in the Sample Input were $2.67.

Charge to user = computer costs + postage and handling + network overhead

= $2.67 + $5.00 + network overhead

= $7.67 + network overhead

CONTENTS—MSA-I

<table>
<thead>
<tr>
<th>Pages</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>Identification &amp; Abstract</td>
</tr>
<tr>
<td>3-4</td>
<td>User Instructions</td>
</tr>
<tr>
<td>5-8</td>
<td>I/O</td>
</tr>
<tr>
<td>9</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>
DESCRIPTIVE TITLE  Simulation Package for University Research and Training
CALLING NAME  SPURT
INSTALLATION NAME  Vogelback Computing Center
Northwestern University
AUTHOR(S) AND AFFILIATION(S)  Gustave J. Rath
Department of Industrial Engineering and Management Sciences
Martin Goldberg
Leonard Weiner
Northwestern University
LANGUAGE  CDC FORTRAN IV
COMPUTER  CDC 6400
PROGRAM AVAILABILITY  Decks and listings presently available
CONTACT  Lorraine Borman, EIN Technical Representative, Vogelback Computing Center, Northwestern University, 2129 Sheridan Road, Evanston, Ill. 60201
Tel.: (312) 492-3682

FUNCTIONAL ABSTRACT

SPURT is a comprehensive package of USASI Standard FORTRAN routines that are designed for use in simulation modelling. These useful routines, ranging from simple to complex, enable the average FORTRAN programmer to employ simulation techniques without having to learn the semantic and syntactic rules of a new programming language.

The SPURT package is made up of six main parts.

I. CLOCK Generation—SPURT1
II. Stochastic Generators—SPURT2
III. Statistical Computations—SPURT3
IV. Analog Simulators—SPURT4
V. List-Processing and Queue-Manipulation—SPURT5
VI. Matrix and Graphical Output—SPURT6

continued
The following is a listing and brief discussion of the various subroutines contained in each of the six S2URT parts.

CLOCK Generation—SPURT1: to implement discrete-time simulation models; to cause events to occur in the proper time sequence.

The CLOCK subroutine consists basically of two lists:
- Master Time List—contains events scheduled to happen in the future.
- Master Time Queue—contains events that could not take place at the time when they were scheduled to and, therefore, have been rescheduled; i.e., they have been blocked and are waiting in a queue.

Events can be stored on either list.

CLOCK recognizes two basic kinds of events:
- Exogenous—those that are external to the user's routine; these are read from Data Cards by the CLOCK.
- Endogenous—those that are internal to the user's routine; these are generated dynamically and then are maintained by the CLOCK.

Stochastic Generators—SPURT2: to generate samples from various probability distributions and to calculate sample values.

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>STOGN1</td>
<td>Permits sampling from a discrete empirical probability distribution defined by the user.</td>
</tr>
<tr>
<td>STOGN2</td>
<td>Enables the user to approximate a continuous distribution by means of a piecewise linear distribution.</td>
</tr>
<tr>
<td>UNIFRM</td>
<td>Permits the user to sample real values from a uniform distribution in a defined interval.</td>
</tr>
<tr>
<td>RANDIN</td>
<td>Provides a uniform distribution of integers in a defined interval.</td>
</tr>
<tr>
<td>NORMAL</td>
<td>Allows the user to obtain a random sample from a normal distribution with given mean and standard deviation.</td>
</tr>
<tr>
<td>NEGEXP</td>
<td>Permits the user to obtain a random sample from the negative exponential distribution.</td>
</tr>
<tr>
<td>POISSN</td>
<td>Provides the user with a random sample from the Poisson distribution.</td>
</tr>
</tbody>
</table>
ERLANG
Provides a random sample from the Erlang distribution

DISCRT
Permits sampling from a step function describing a discrete cumulative distribution of integer values

LINEAR
Provides the user with a random sample from a cumulative distribution that is obtained by linear interpolation in a nonequidistant table of real values

DRAW
Provides a boolean value of TRUE or FALSE

RANPER
Generates a uniformly distributed, random permutation of the integers 1, 2, ..., M

Statistical Computations—SPURT3: to calculate statistical parameters and histograms of data arrays

Subroutine  Usage
STIX1  Three interrelated subroutines to accumulate and print out a frequency table and to produce a CalComp plot of a normalized histogram of the table
STIX3  Evaluates the mean, standard deviation, maximum value, and minimum value of an array of real numbers
STIX4  Evaluates the correlation coefficient between two arrays of real numbers
STIX5  Ranks an array of real numbers and produces the median and range of the data within the array
STIX6  Produces a statistical description of the data found in an array, including the sample size, mean, standard deviation, standard error, minimum and maximum values, range, and a printed histogram plot

Analog Simulators—SPURT4: to enable the simulation of analog-computer problems on a digital computer

Subroutine  Usage
ANALOG  These two subroutines make it possible to obtain output similar to a hybrid computer
List-Processing and Queue-Manipulation—SPURT5: lists are mXn arrays; entries in lists are mX1 arrays.

**Subroutine Usage**

**ADFIFO**
Adds an entry at the bottom of the list; it can be removed only after all the elements presently on the list are gone (builds first-in last-out list)

**ADLIFO**
Adds an entry at the top of the list; it will be removed before any other entry presently on the list (builds last-in first-out list)

**REMOVE**
Removes the top (or first) entry from a list

**PURGE**
Destroys the contents of a list

**DISPL**
Prints the contents of a list

Additional subroutines in SPURT5 provide the capability to rank lists and to delete or to insert entries into lists.

Matrix and Graphical Output—SPURT6: output is facilitated through printing and graphical output

**Subroutine Usage**

**OUT**
Prints out a square matrix with column and row headings

**NSOUT**
Prints out a nonsquare matrix with column and row headings

**GRAPH**
Produces two-dimensional graphs of plots, using a CalComp plotter

REFERENCES

USER INSTRUCTIONS

SPURT is separated into six sections (Parts) in the Vogelback Program Library. All of the routines contained in any one Part will be loaded with the user's program when he places that Part name in the argument list of a Library Card. The following listing shows which routines are included in each part of SPURT.

<table>
<thead>
<tr>
<th>Program Library Name (Part)</th>
<th>SPURT1</th>
<th>SPURT2</th>
<th>SPURT3</th>
<th>JRT4</th>
<th>SPURT5</th>
<th>SPURT6</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPURT User Routines Loaded:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIRCLESY</td>
<td>STOGN1</td>
<td>OUT</td>
<td>ANALOG</td>
<td>ADLIFO</td>
<td>PAGE</td>
<td></td>
</tr>
<tr>
<td>NORMALL</td>
<td>STOGN2</td>
<td>IOUT</td>
<td>SECND</td>
<td>REMOVE</td>
<td>DIAGS</td>
<td></td>
</tr>
<tr>
<td>SECNDF</td>
<td>UNIFRT</td>
<td>INSOU</td>
<td>DISPCL</td>
<td>PURGE</td>
<td>DIAGNW</td>
<td></td>
</tr>
<tr>
<td>RANDINF</td>
<td>GRAPH</td>
<td>INS</td>
<td>NUMS</td>
<td>DISPCL</td>
<td>SIZS</td>
<td></td>
</tr>
<tr>
<td>NORMALS</td>
<td>STIX1</td>
<td>DISPCL</td>
<td>INRKS</td>
<td>SUMS</td>
<td>RANKQL</td>
<td></td>
</tr>
<tr>
<td>NEGEXP</td>
<td>STIX2</td>
<td>NUMST</td>
<td>INRKS</td>
<td>RANKQL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>POISSN</td>
<td>STIX3</td>
<td>RANKQ</td>
<td>INRKS</td>
<td>RANKQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERLANG</td>
<td>STIX4</td>
<td>INRKQ</td>
<td>SIZS</td>
<td>RANKQL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DISCRT</td>
<td>STIX5</td>
<td>INRKQ</td>
<td>RANKQL</td>
<td>RANKQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LINEAR</td>
<td>STIX6</td>
<td>INRKQ</td>
<td>RANKQL</td>
<td>RANKQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DRAW</td>
<td>STIX7</td>
<td>INRKQ</td>
<td>RANKQL</td>
<td>RANKQ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HISTD</td>
<td>INSERT</td>
<td>NSOUT</td>
<td>REMOV</td>
<td>DIAGNS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RANPER</td>
<td>DELETE</td>
<td>NZOUT</td>
<td>RANPER</td>
<td>DIAGS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| SPURT Internal Routines Loaded: |        |        |        |      |        |        |
| ZZZXXQ9Q | QRLQPB |        |        |      |        |        |
| LODLST | PURGER |        |        |      |        |        |

Common Blocks Used:

| CIRCLESY | SPRTFT | SIZS | Qiqueta |
| NORMALL | SPRTIO | SIZS | Qيقة |
| SECNDF | SIZS | Qيقة | SPTMSG |
| RANDINF | Qيقة | SPTMSG | SPTMSG |
| NORMALS | SPRTIO | SPTMSG | SPTMSG |
| NEGEXP | SPRTFT | SPTMSG | SPTMSG |
| POISSN | Qيقة | SPTMSG | SPTMSG |
| ERLANG | SPTMSG | SPTMSG | SPTMSG |
| DISCRT | SPTMSG | SPTMSG | SPTMSG |
| LINEAR | SPTMSG | SPTMSG | SPTMSG |
| DRAW | SPTMSG | SPTMSG | SPTMSG |
| HISTD | INSERT | NSOUT | REMOV | DIAGNS | |
| RANPER | DELETE | NZOUT | RANPER | DIAGS | |

| SPURT Part Core Requirements (octal): | 6500 | 4000 | 21300 + 6400 | 5700 | 5000 |
| Part Core Requirements When Loaded with SPURT6 (octal): | 7600 | 6600 | 22500 + 7600 | 7100 | — |

continued

2/70 5
NOTE: Whenever any of the SPURT routines is called from the library, SPURT6 must also be called. Therefore, every use of the Library Card for SPURT will include SPURT6 in the argument list.

Application Hints

All user-written subroutines should have as their first executable instruction a call to the user-written subroutine TRACE, passing as a parameter the name of the calling subroutine. For example, the first executable statement in subroutine FACTORY, say, should be CALL TRACE (FACTORY). Thus, as each subroutine within the simulation model initiates execution, it will call on the TRACE routine, which can be constructed to print a short message giving the current simulated time, the name of the calling subroutine, and the status of key system variables.

It should be emphasized that TRACE is a user-written closed subroutine. Therefore, it is the user's responsibility to construct the printout to include the simulation time, calling subroutine name, and all key system variables.

Segment coding is suggested because discrete-time simulation models are characterized by a chain of events and/or activities. Each event and activity is well defined in setting up the problem solution. This implies that each event or activity within the model should be coded as a closed subroutine. In doing this, it is good practice to place into a common block all variables used by more than one routine. Segmentation of this sort will also allow the easy generation of closed subroutines to handle data input, statistical analysis, printouts, and facilities for a TRACE routine.

Error-Indicator Subroutines

DIAGNS and/or DIAGNW

SUBROUTINE DIAGNS (NAME,NUMBR)
SUBROUTINE DIAGNW (NAME,NUMBR)

where NUMBR is an integer and NAME is a Hollerith constant.

The user may wish to incorporate calls to DIAGNW and/or DIAGNS in his main program. These are called by most of the SPURT subroutines and functions whenever an error condition arises during a run. If the error is of a serious nature, DIAGNS is called; it causes a program termination after an appropriate diagnostic message has been printed on the standard output unit. For less serious errors, DIAGNW is called by the SPURT routine detecting the error; control is returned to the calling routine after DIAGNW.

continued
prints out a message. (See Ref. 1 for a listing of all diagnostic messages.)

The user is given the ability to force SPURT to continue execution of a program after detecting a normally fatal SPURT error condition during the run. To continue, the user must preset the third word of COMMON/SZSYSP to nonzero in his main program. This causes all calls (by SPURT) to DIAGNS to be handled as calls to DIAGNW.

CAUTION: The disablement of the termination feature of DIAGNS will merely cause an incorrect value to be stored in X; however, with many other SPURT routines (e.g., CLOCK), infinite loops may occur when this feature is disabled. This option is to be used with extreme care!

REFERENCES

USER INSTRUCTIONS—SPURT1: CLOCK Generation

SUBROUTINE CLOCK (L,NEWT,NEWJ)
SUBROUTINE SETCLK (MTL,JUMP,LMTL,MTQ,JUMQ,LMTQ)
TYPE INTEGER L,NEWT,NEWJ,MTL,JUMP,LMTL,MTQ,JUMQ,LMTQ

The primary function of CLOCK is the manipulation of two lists: a Master Time List and a Master Time Queue. Events and their type number and scheduled time of occurrence can be stored on either list. The Master Time List contains events that are scheduled to happen; the Master Time Queue contains events that could not take place at the time when they were originally scheduled to and, therefore, have been rescheduled. That is, they were blocked and so are waiting in a queue. CLOCK causes events to occur at the proper simulation time. (Unless otherwise noted, all references to time here will mean simulated time.)

The Master Time List consists of two one-dimensional arrays: MTL and JUMP. Both are of the same size and must be dimensioned in the user's program. The MTL array contains the time at which the events are scheduled to occur. The JUMP array contains the number of the event type. The ith element of MTL and the ith elements of JUMP constitute an event. LMTL stands for the length of the Master Time List (LMTL < that of MTL). NEVEN is the variable that contains the current number of events in the Master Time List.

NOTE: The value of NEVEN is maintained by the CLOCK and should not be changed by the user. It is intended for use on a read-only basis.

The Master Time Queue is very similar to the Master Time List, except that it contains blocked events. The variables MTQ, JUMQ, LMTQ, and NEVEQ correspond exactly to the variables MTL, JUMP, LMTL, and NEVEN for the Master Time List. Thus, MTQ is a one-dimensional array containing event types. Both arrays are dimensioned by the user in his program and must have the same dimension. Also, LMTQ must be defined (LMTQ < dimension of MTQ). NEVEQ is the current number of events in the Master Time Queue.

CLOCK recognizes two basic kinds of events: exogenous (external to the user's program and the CLOCK subroutine) and endogenous (internal to the program).

Exogenous events are read in on data cards as input data, one event per card. The data cards must be arranged by the user in time sequence. To have SETCLK or CLOCK read an exogenous event, the user must specify the logical unit number (LUN) from which the cards are

continued
to be read. If the data cards follow the program on the standard system input, LUN = 5. If the value for LUN is not set between 1 and 99, the reading of exogenous events is suppressed.

Endogenous events are not read in on data cards and are created in the user's program.

The basic function of CLOCK is to cause events to happen when they are scheduled. When CLOCK is initialized, the first exogenous event is read (if a valid LUN has been specified) and is placed on the Master Time List. Subsequently, whenever the normal CALL CLOCK statement is used, the subroutine causes the event to be taken from the Master Time List or Master Time Queue, whichever has the earliest time of occurrence (ties are decided in favor of the Master Time Queue) and returned to the user's program. Whenever an exogenous event is deleted from the Master Time List, the next exogenous event is automatically read (if LUN is valid) and placed on the Master Time List in sequence. CLOCK works with only one event at a time and stops when all exogenous events have been read in and both lists are empty.

The user initializes CLOCK with the following call: CALL SETCLK(MTL, JUMP, LMTL, MTQ, JUMQ, LMTQ). The parameters in the calling sequence are the same as those described above.

If the user wishes to read an exogenous event, then common block CLOCK1 must be accessed (see Ref. 1, p. 3.9) and LUN preset equal to 5. (see Ref. 1, p. 3.4).

The user calls CLOCK with CALL CLOCK(L, NEWT, NEWJ). The only variables that will change in the course of the user's program are L, NEWT, and NEWJ. Thus, SETCLK is called only when it is desired to initialize both clock lists. Here, NEWT contains the time of the event and NEWJ contains the number of the event type. Since CLOCK works with only one event at a time, NEWT and NEWJ will always contain the time and type number of the present event being worked with. Each event type is considered to be the parameter of a computer GO TO statement in the user's program and represents the jump address of an event to be executed (Ref. 1, Appendix).

The variable L takes on integer values from 2 through 9; the value must be chosen by the user and determines the function of the call statement. Example: Consider the use of L in the following calls to the CLOCK subroutine.

L = 2: used for loading the Master Time List with the event (NEWT, NEWJ)
CALL CLOCK(2,NEWT,NEWJ)

The parameter value of L equal to 2 denotes that the Master Time List is to be loaded with event type NEWJ that is scheduled to occur at time NEWT.

L = 3: used to delete from both lists all events of a given type NEWJ occurring at a given time NEWT—

CALL CLOCK(3,NEWT,NEWJ)

This call to CLOCK is used to delete the occurrence of event type NEWJ at a scheduled time of NEWT from both the Master Time List and the Master Time Queue. For example, CALL CLOCK(3,8,1) requests that event (8,1) be deleted from both lists.

L = 4: used for calling the next scheduled event—

CALL CLOCK(4,NEWT,NEWJ)

Here, a request has been made of the CLOCK for the next scheduled event and its time of occurrence. CLOCK returns a value in NEWT equal to the time of occurrence of the next event type. The value of the event-type code is returned in NEWJ.

L = 5: used for loading a blocked event (NEWT,NEWJ) on the Master Time Queue—

CALL CLOCK(5,NEWT,NEWJ)

A blocked event (NEWT,NEWJ) is to be loaded onto the Master Time Queue. The user is cautioned that NEWT must not contain the time at which the event was blocked. If this is included, the clock will hang in an endless loop. NEWT is the next time of occurrence of the event type NEWJ after it has been blocked.

L = 6: produces the next scheduled time of occurrence (i.e., NEWT for an event of type NEWJ—

CALL CLOCK(6,NEWT,NEWJ)

This call to the CLOCK requests the next scheduled time of occurrence of event type NEWJ. The answer to the request is returned in NEWT. For example, CALL CLOCK(6,NEWT,4) will return to the user (in NEWT the next time of occurrence of event type 4. If a request for the next scheduled time of occurrence of event type NEWJ cannot be satisfied (e.g., if that event type is not on the CLOCK lists), then the value returned in NEWT is negative.

continued
L = 7: purges (deletes) both lists of all events of type NEWJ—
CALL CLOCK(7,0,NEWJ)

Here, the CLOCK is requested to delete from both the Master Time
List and the Master Time Queue all occurrences of the event type
specified by NEWJ. Note that the second parameter (NEWT) is de-
noted as 0, as this parameter is ignored by this request but must
be filled in the calling sequence.

L = 8: purges the Master Time List of all events of type NEWJ—
CALL CLOCK(8,0,NEWJ)

This request is similar to 7, except that it deletes all events of
type NEWJ from the Master Time List only.

L = 9: purges the Master Time Queue of all events of type NEWJ—
CALL CLOSK(9,0,NEWJ)

This request is similar to 7 also, except that it deletes all events
of type NEWJ from the Master Time Queue only.

NOTE: If an event is blocked for some reason, the user must place
it on the Master Time Queue, at a later scheduled time, by using a
different form of the CALL CLOCK (Ref. 1, p. 3.4).

The unit in which event times are measured is decided by the user
in terms of the process being simulated. However, event times must
be nonnegative integers.

The user has access to the values of NEVEN, NEVEQ, and LUN, used by
the CLOCK. These values are placed in the labeled common block
CLOCK1. (See Ref. 1, p.3.9.) NEVEN and NEVEQ should be accessed
only on a read-only basis by the user. CLOCK sets and updates both
variables.

If the user wants to run a real-time simulation, he can use FUNCTION
TIMELP, which provides the elapsed CP time between successive calls.

TIMELP
REAL FUNCTION TIMELP (NPRINT)
TYPE INTEGER NPRINT

If the user wants to attempt a simulation in real time, he may find
FUNCTION TIMELP useful. This routine prints (at the user's option)
and returns the CDC 6400 central processor elapsed time (in seconds)
between consecutive calls to TIMELP. An initializing call to TIMELP should be made at the beginning of execution of the routine to be timed. TIMELP can also be used to time any particular portion of a FORTRAN program.

A zero value of NPRINT suppresses printing by TIMELP. A nonzero NPRINT will print the legend CP TIME SINCE LAST CALL .. xx.xxx SECONDS. In either case, TIMELP contains the elapsed time. The call to TIMELP takes the form CALL TIMELP (NPRINT).

Job Deck Structure

Job Cards
Run(s)
LIBRARY(SPURT1,SPURT6)
LGO.
7-8-9 (EOR Card)
Main-Problem Card(s)
PROGRAM(INPUT,OUTPUT,PUNCH, PLOT, TAPE5=INPUT, 1 TAPE6=OUTPUT,
   TAPE7=PUNCH, TAPE17=PLOT:
   .
   .
   END
Subroutine Card(s)
   SUBROUTINE ACTION1
   .
   .
   END
   .
   .
   SUBROUTINE ACTION2
   .
   .
   END
7-8-9 (EOR Card)
Data Cards
6-7-8-9 (EOI Card)

NOTE: The file assignments on the Main-Program header card are the standard SPURT assignments.
**SAMPLE INPUT**

SUBROUTINE EXAMPL3

** THIS PROGRAM IS PART OF THE NORTHWESTERN UNIVERSITY **

**SPURT**

** A SIMULATION PACKAGE FOR UNIVERSITY RESEARCH AND TEACHING **

******************************************************************************

COMMON /CLOCK/ DUMMY,NEVEN,NUMMT,NEVEN,LUN
COMMON MTL(500), JUMP(500), MTO(100), JUM0(100)

** THE FOLLOWING CARDS ARE COPIES OF THE EXOGENOUS EVENTS TO BE READ **

** NEWT NEWJ **

2 5
5 1
24 8

** END OF EXOGENOUS EVENTS LIST **

** INITIALIZATION **

LMTL=5000,
LMT0=100
L1=1
MMJ,J=1
LUNM=
DUMATHELP(0)
CALL SETCLK (MTL,JUMP,MTL,JMP,JMT,JMT0)
DUMATHELP(1)
PRINT 140
PRINT 130,
DUMATHELP(1)
CALL 00 NEXT EVENT

** DUMMY ROUTINE TO LOAD ENDOGENOUS EVENTS ON MASTER TIME LIST **

L4=20 IN1,10
F=1
JJJ (ABS(COS(F)))*T,0,1,0
MM=2/1
CALL CLOCK (L1,MM,JJ)
PRINT 130,
DUMATHELP(1)
CALL OUT NEXT EVENT

** CALL CLOCK (L1,MM,JJ) **

PRINT 130,
DUMATHELP(1)
GO TO (100+200,300+400+500+800+700+800),JJ

K=100
PRINT 130, K,MM
GO TO 30

continued
PRINT 120, K:MM
L=7
CALL C i C L (L,MM,JJ)
PRINT 130, HH, JJ, L + NEVEN, NEVEQ
DUMP TIME(L,1)
GO TO 30

C
C

FORMAT (1WO, ENTITY, *I5.8, *ACTIVATED AT SIMULATION TIME = **I8) SPTS9230
FORMAT (1W0, 3DX, 18, I4X, I8, I8, I8, I8, I8, I8) SPTS9230
FORMAT (3DX, *SIMULATED TIME = I0X*JUMP ADDR *10X, SPTS9240
1 *16X*NEVEN, 16X*NEVES***/) SPTS9240
END
ERROR ENCOUNTERED IN ROUTINE...CLOCK

REQUEST FOR NEXT SCHEDULED TIME OF OCCURRENCE OF EVENT UNSATISFIED... EVENT NOT ON LISTS

PROGRAM CONTINUING...

ERROR ENCOUNTERED IN ROUTINE...CLOCK

BOTH CLOCK LISTS EXHAUSTED.

JOB TERMINATED.
USER INSTRUCTIONS—SPURT2: Stochastic Generators

The STOGN (stochastic-generating) functions enable the user to generate samples from various user-defined probability distributions and calculate sample values.

All of the STOGN functions that produce random values use the standard library random-number-generating function RANF. Unless otherwise preset, the seed (and, therefore, the entire stream of numbers generated) is the same for each run. To change the seed, the user should CALL RANSET (SEED) during his simulation system initialization. A convenient value for SEED may be determined by SEED = TIME(0). This should produce a new seed each time.

NOTE: The user is cautioned to use a TYPE statement in all of his routines that use the stochastic generators. Failure to specify the routine name as a TYPE INTEGER could result in improper evaluation of mixed-mode expressions.

STOGN1

REAL FUNCTION STOGN1(NN,XXX,ORD,K)
TYPE REAL XXX, ORD
TYPE INTEGER NN, K

STOGN1 is a discrete, empirical, probability distribution, defined by the user. This FORTRAN function accesses a user-defined probability distribution that is called by VALUE = STOGN1 (NN,XXX,ORD,K). For each empirical probability distribution that the user employs, he must define two two-dimensional arrays XXX and ORD, with the 2nd subscript of these two arrays being the identification number that the user assigns to the probability distribution. K is the name of the probability distribution. Any allowable variable names of TYPE REAL may be substituted for XXX and ORD. The arrays must be dimensioned in the user's program, along with defining the variable NN.

STOGN2

REAL FUNCTION STOGN2(NN,XXX,ORD,K)
TYPE REAL XXX, ORD
TYPE INTEGER NN, K

STOGN2 enables the user to approximate a continuous probability distribution by means of a piecewise linear one. The statement VALUE = STOGN2(NN,XXX,ORD,K) will cause to be returned values that represent a random sample from a particular distribution. Otherwise, the same rules apply to the general case for STOGN2 as for STOGN1.
STOGN6

REAL FUNCTION STOGN6(N,P)
TYPE REAL P
TYPE INTEGER N

STOGN6 enables the user to obtain a random sample from the negative binomial distribution, with parameters N (number of successful trials) and P (probability of success). S is the value returned by STOGN6 and represents the number of unsuccessful trials. The call is of the form S = STOGN6(N,P). The user is cautioned to observe the restrictions of N ≥ 0 and 0.0 < P < 1.0.

UNIFRM, NORMAL

REAL FUNCTION UNIFRM(A,B)
REAL FUNCTION NORMAL(A,B)
TYPE REAL A,B

UNIFRM obtains a sample of real values that are equally likely to fall between the values A and B, where A < B. The user is able to sample from a uniform distribution defined on the interval from A to B. The function is called by XNUM = UNIFRM(A,B).

NORMAL allows the user to obtain a random sample from a normal distribution, with mean A and standard deviation B. FUNCTION NORMAL is called by XNUM = NORMAL(A,B).

RANDIN

INTEGER FUNCTION RANDIN(A,B)
TYPE INTEGER A,B

RANDIN(A,B) provides a uniform distribution of integers between the limits of A and B. The value is one of the integers A, A + 1, ⋯, B − 1, B, where A < B, with equal probability of each. The call to this function is of the form NUM = RANDIN(A,B).

LINEAR, DRAW

REAL FUNCTION LINEAR(A,B)
LOGICAL FUNCTION DRAW(A)
TYPE REAL A,B

LINEAR provides the user with a random supply from a (cumulative) distribution function F, which is obtained by linear interpolation in a nonequidistant table defined by A and B, such that A(i) = F(B(i)). It is assumed that A and B are one-dimensional real arrays of the same length, that the first and last elements of A are equal to 0.0

continued
and 1.0, respectively, and that \( A(i) < A(j) \) and \( B(i) < B(j) \) for \( i < j \). The call to LINEAR is \( XNUM = \text{LINEAR}(A,B) \).

DRAW returns a value of .TRUE., with the probability \( A \) and .FALSE. with the probability \( 1 - A \). DRAW is always .TRUE. if \( A > 1 \) and is always .FALSE. if \( A < 0 \). Consider the following example of the use of the function DRAW: IF (DRAW(0.72)) 20, 30. This will transfer control to statement 20 with a probability of occurrence of 0.72 and to statement 30 with a probability of 0.28.

NEGEXP, ERLANG

REAL FUNCTION NEGEXP(A)
REAL FUNCTION ERLANG(A,B)
TYPE REAL A,B

NEGEXP obtains a random sample from the negative exponential distribution with mean \( 1/A \), defined by \(-1/A [\ln(\text{random number})]\). This is the same as random "waiting time" in a Poisson-distributed arrival pattern with expected number of arrivals per unit time equal to \( A \). A call to FUNCTION NEGEXP could be \( XNUM = \text{NEGEXP}(A) \) or \( XNUM = \text{NEGEXP}(1.0/XMEAN) \).

ERLANG obtains a random sample from the Erlang distribution with mean \( 1/A \) and standard deviation \( 1/(A/B) \). It is defined by \( B \) basic drawings of a random number, if \( B \) is an integer value, or, otherwise, by \( B + 1 \) drawings. Both \( A \) and \( B \) must be greater than zero in the call to ERLANG: \( XNUM = \text{ERLANG}(A,B) \).

POISSN

INTEGER FUNCTION POISSN(A)
TYPE REAL A

POISSN provides a random sample from the Poisson distribution with parameter \( A \). It is obtained by \( n + 1 \) basic drawings of random numbers, where \( n \) is the function value defined as the smallest non-negative integer for which

\[
\prod_{i=0}^{n} \text{RANF}(0) < e^{-A},
\]

where \( \text{RANF} \) is the random-number-generating function. When the parameter of \( A \) is greater than 20.0, the value is approximated by the integer value of \( \text{NORMAL}(A, \text{SQRT}(A)) \) or, when this is negative, by zero. A sample call is of the form \( \text{NUM} = \text{POISSN}(A) \).
DISCRT
INTEGER FUNCTION DISCRT(A)
TYPE REAL A

The one-dimensional real array A is interpreted as a step function of the subscript, defining a discrete (cumulative) distribution function. The function value is an integer in the range \([\text{lsb} - 1, \text{usb} - 1]\), where \(\text{lsb}\) and \(\text{usb}\) are the lower and upper subscript bounds of the array. It is defined as the smallest \(i\), such that \(A(i + 1) > \mu\), where \(\mu\) is the basic random number produced by RANF and where \(A(\text{usb}) = 1.0\). The call is of the form \(\text{NUM} = \text{DISCRT}(A)\).

HISTD
INTEGER FUNCTION HISTD(A)
TYPE REAL A

HISTD provides the user with the random value of an integer in the range \(\text{[lsb, usb]}\), where \(\text{lsb}\) and \(\text{usb}\) are the lower and upper subscript bounds of the one-dimensional real array A. The latter is interpreted as a histogram defining the relative frequencies of the values.

This function is more time-consuming than the function DISCRT, where the cumulative distribution function is given, but it is more useful if the frequency histogram is dynamically updated at run time. A sample call is \(\text{NUM} = \text{HISTD}(A)\).

RANPER
SUBROUTINE RANPER(N,M)
TYPE INTEGER N,M

Subroutine RANPER generates a uniformly distributed, random permutation of the integers 1, 2, \ldots, \(M\) and stores it in the array \(N\), where \(N\) is an integer-valued array of dimension \(M(>1)\). RANPER is called by CALL RANPER(N,M).
Job Deck Structure

Job Cards
Run(s)
LIBRARY(Spurt1, Spurt6)
LGO.
7-8-9 (EOR Card)

Main-Problem Cards

```
PROGRAM(INPUT,OUTPUT,PUNCH,PLOT,TAPE5=INPUT, TAPE6=OUTPUT,
          TAPE7=PUNCH,TAPE17=PLOT)

```

7-8-9 (EOR Card)

Subroutine Card(s)

```
SUBROUTINE ACTION1

```

```
SUBROUTINE ACTIONZ

```

7-8-9 (EOR Card)

Data Cards

```
6-7-8-9 (EOI Card)

```

NOTE: The file assignments shown on the Main-Program header card are the standard Spurt assignments.
SAMPLE INPUT

SUBROUTINE TESTGEN

******************************************************************************

THIS PROGRAM IS PART OF THE NORTHWESTERN UNIVERSITY

SPURT

A SIMULATION PACKAGE FOR UNIVERSITY RESEARCH AND TEACHING

******************************************************************************

THIS IS A TEST PROGRAM TO CHECK THE RESULTS OF THE
THE RANDOM NUMBER GENERATORS OF THE SPURT PACKAGE
AND PLOT PACKAGE (STIX1,STIX2,STIX3,GRAPH,ENPLT)

******************************************************************************

INTEGER RANDIN, POISSNOIST, DO, SCRT

DIMENSION XA(4)

COMMON /PLOTSC/ A(5), PLOTSW

DIMENSION XXX1(3, 2), ORD2(13, 2), XXX1(5, 1), ORD1(5, 1)

COMMON VALUES1(104), VALUES2(104), VALUES3(104), VALUES4(104), VALUES5(104), VALUES6(104)

DATA (0.2, 0.4, 1.0), (0.5, 0.1, 0.6), (0.9, 0.5, 0.4)

DO 10 1, 10000

CALL STIX1(VALUES20.0, 3, 0.4, VALUE, STOGN1(3, XXX2, ORD2, 1),

CALL STIX2(VALUES2.7, 0.0, 4.0, VALUE, STOGN1(3, XXX2, ORD2, 1),

1 CONTINUE

CALL STIX1(VALUES1, 4.0, 0.0, 4.0, VALUE, STOGN1(3, XXX2, ORD2, 1),

DO 30 1, 10000

CALL STIX2(VALUES140.0, 0.0, 4.0, VALUE, STOGN1(3, XXX2, ORD2, 1),

30 CONTINUE

continued
CALL STIX1(VALUES1,21.0,21.6,42,VALUE,6HRANDIN(6))
CALL STIX2(VALUES1,21.0,21.6,42,VALUE,6HRANDIN(6))
CALL STIX3(VALUES1,21.0,21.6,42,VALUE,6HRANDIN(6))
DUMATMELP(1)
DO 106 181,10000
VALUE^NORMAL(0.0,1001
CALL JTIX2(VALUES194,09.4,0,46,VALUE,6HPOISSN(4))
DO 116 1°1910000
VALUEEPOISSN(5,0)
CALL STIX2(VALUES1,10.0,0,0.46,VALUE,6HERLANG(6))
DUMATIMELP(1)
CALL STIX1(VALUES1,10.0,0,0.46,VALUE,6HERLANG(6))
GO TO 166
END
Entries in table

<table>
<thead>
<tr>
<th>Mean Argument</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000</td>
<td>7.05</td>
</tr>
</tbody>
</table>

Entries in table

<table>
<thead>
<tr>
<th>Upper Limit</th>
<th>Observed Frequency</th>
<th>Per Cent of Total</th>
<th>Cumulative Percentage</th>
<th>Cumulative Remainder</th>
<th>Multiple of Mean</th>
<th>Deviation from Mean</th>
<th>Cumulative Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.000</td>
<td>0</td>
<td>0.000</td>
<td>100.000</td>
<td>0.000</td>
<td>0.688</td>
<td>-3.001</td>
<td>0.001</td>
</tr>
<tr>
<td>3.100</td>
<td>0</td>
<td>0.000</td>
<td>100.000</td>
<td>0.000</td>
<td>0.698</td>
<td>-2.956</td>
<td>0.002</td>
</tr>
<tr>
<td>3.200</td>
<td>0</td>
<td>0.000</td>
<td>100.000</td>
<td>0.000</td>
<td>0.627</td>
<td>-2.716</td>
<td>0.001</td>
</tr>
<tr>
<td>3.300</td>
<td>0</td>
<td>0.000</td>
<td>100.000</td>
<td>0.000</td>
<td>0.667</td>
<td>-2.573</td>
<td>0.006</td>
</tr>
<tr>
<td>3.400</td>
<td>0</td>
<td>0.000</td>
<td>100.000</td>
<td>0.000</td>
<td>0.667</td>
<td>-2.430</td>
<td></td>
</tr>
</tbody>
</table>

Graph number 1 has just been plotted on calcomp

The plot is of...

Stogoni

vs.

frequency (per cent)

End of output by graph

CP time since last call 5.395 seconds
### Normal Distribution

**Entries in Table 10000**

<table>
<thead>
<tr>
<th>Mean Argument</th>
<th>Standard Deviation</th>
<th>Normal Entries in Table 10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>-10.00</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td>-9.00</td>
<td>1</td>
<td>0.010</td>
</tr>
<tr>
<td>-8.00</td>
<td>2</td>
<td>0.020</td>
</tr>
<tr>
<td>-7.00</td>
<td>3</td>
<td>0.030</td>
</tr>
<tr>
<td>-6.00</td>
<td>4</td>
<td>0.040</td>
</tr>
<tr>
<td>-5.00</td>
<td>5</td>
<td>0.050</td>
</tr>
<tr>
<td>-4.00</td>
<td>6</td>
<td>0.060</td>
</tr>
<tr>
<td>-3.00</td>
<td>7</td>
<td>0.070</td>
</tr>
<tr>
<td>-2.00</td>
<td>8</td>
<td>0.080</td>
</tr>
<tr>
<td>-1.00</td>
<td>9</td>
<td>0.090</td>
</tr>
<tr>
<td>0.00</td>
<td>10</td>
<td>0.100</td>
</tr>
<tr>
<td>1.00</td>
<td>11</td>
<td>0.110</td>
</tr>
<tr>
<td>2.00</td>
<td>12</td>
<td>0.120</td>
</tr>
<tr>
<td>3.00</td>
<td>13</td>
<td>0.130</td>
</tr>
<tr>
<td>4.00</td>
<td>14</td>
<td>0.140</td>
</tr>
</tbody>
</table>

**Percent of Total**

- Cumulative Probability
- Cumulative Deviation
- Cumulative Multiple

**Upper Limit**

- 0.000
- 1.000
- 2.000
- 3.000
- 4.000
- 5.000
- 6.000
- 7.000
- 8.000
- 9.000
- 10.000

**Observed Frequency**

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

**Percent of Total**

- Cumulative Probability
- Cumulative Deviation
- Cumulative Multiple

**Observed Frequency**

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

**Cumulative Probability**

- 0.000
- 0.010
- 0.020
- 0.030
- 0.040
- 0.050
- 0.060
- 0.070
- 0.080
- 0.090
- 0.100

**Cumulative Deviation**

- 0.000
- 0.010
- 0.020
- 0.030
- 0.040
- 0.050
- 0.060
- 0.070
- 0.080
- 0.090
- 0.100

**Cumulative Multiple**

- 0.000
- 0.010
- 0.020
- 0.030
- 0.040
- 0.050
- 0.060
- 0.070
- 0.080
- 0.090
- 0.100

**Cumulative Probability**

- 0.000
- 0.010
- 0.020
- 0.030
- 0.040
- 0.050
- 0.060
- 0.070
- 0.080
- 0.090
- 0.100
<table>
<thead>
<tr>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
<th>C7</th>
<th>C8</th>
<th>C9</th>
<th>C10</th>
</tr>
</thead>
<tbody>
<tr>
<td>369</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
<tr>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
<td>361</td>
</tr>
</tbody>
</table>

This data is accompanied by graph number 999.
### HISTO

<table>
<thead>
<tr>
<th>ENTRIES IN TABLE</th>
<th>MEAN ARGUMENT</th>
<th>STANDARD DEVIATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upper Limit</td>
<td>2-500</td>
<td>0.91</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNDERFLOW</th>
<th>OBSERVED FREQUENCY</th>
<th>PER CENT OF TOTAL</th>
<th>CUMULATIVE PERCENTAGE</th>
<th>CUMULATIVE REMAINDER</th>
<th>MULTIPLE OF MEAN</th>
<th>DEVIATION FROM MEAN</th>
<th>CUMULATIVE PROBABILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;500</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>100.000</td>
<td>2.000</td>
<td>-2.167</td>
<td>0.014</td>
</tr>
<tr>
<td>&lt;600</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>100.000</td>
<td>2.000</td>
<td>-2.098</td>
<td>0.019</td>
</tr>
<tr>
<td>&lt;700</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>100.000</td>
<td>2.000</td>
<td>-1.960</td>
<td>0.025</td>
</tr>
<tr>
<td>&lt;800</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>100.000</td>
<td>2.000</td>
<td>-1.869</td>
<td>0.032</td>
</tr>
<tr>
<td>&lt;900</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>100.000</td>
<td>1.960</td>
<td>-1.760</td>
<td>0.040</td>
</tr>
<tr>
<td>&lt;1000</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>100.000</td>
<td>1.940</td>
<td>-1.640</td>
<td>0.050</td>
</tr>
<tr>
<td>1.000</td>
<td>1471</td>
<td>14.710</td>
<td>14.710</td>
<td>85.290</td>
<td>1.400</td>
<td>1.640</td>
<td>0.063</td>
</tr>
<tr>
<td>1.000</td>
<td>1461</td>
<td>14.610</td>
<td>14.610</td>
<td>85.390</td>
<td>1.400</td>
<td>1.640</td>
<td>0.063</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OVERFLOW</th>
<th>OBSERVED FREQUENCY</th>
<th>PER CENT OF TOTAL</th>
<th>CUMULATIVE PERCENTAGE</th>
<th>CUMULATIVE REMAINDER</th>
<th>MULTIPLE OF MEAN</th>
<th>DEVIATION FROM MEAN</th>
<th>CUMULATIVE PROBABILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;4000</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>85.390</td>
<td>1.400</td>
<td>1.640</td>
<td>0.049</td>
</tr>
<tr>
<td>&lt;4100</td>
<td>1461</td>
<td>14.610</td>
<td>14.610</td>
<td>85.390</td>
<td>1.400</td>
<td>1.640</td>
<td>0.049</td>
</tr>
<tr>
<td>&lt;4200</td>
<td>1460</td>
<td>14.600</td>
<td>14.600</td>
<td>85.390</td>
<td>1.400</td>
<td>1.640</td>
<td>0.049</td>
</tr>
<tr>
<td>&lt;4300</td>
<td>1460</td>
<td>14.600</td>
<td>14.600</td>
<td>85.390</td>
<td>1.400</td>
<td>1.640</td>
<td>0.049</td>
</tr>
<tr>
<td>&lt;4400</td>
<td>1460</td>
<td>14.600</td>
<td>14.600</td>
<td>85.390</td>
<td>1.400</td>
<td>1.640</td>
<td>0.049</td>
</tr>
<tr>
<td>&lt;4500</td>
<td>1460</td>
<td>14.600</td>
<td>14.600</td>
<td>85.390</td>
<td>1.400</td>
<td>1.640</td>
<td>0.049</td>
</tr>
</tbody>
</table>

### INSCUT

<table>
<thead>
<tr>
<th>DATE 01/27/70</th>
<th>PAGE NO. 92</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1 11</td>
<td>C2 12</td>
</tr>
<tr>
<td>C3 13</td>
<td>C4 14</td>
</tr>
<tr>
<td>C5 15</td>
<td>C6 16</td>
</tr>
<tr>
<td>C7 17</td>
<td>C8 18</td>
</tr>
<tr>
<td>C9 19</td>
<td>CI0 20</td>
</tr>
</tbody>
</table>
USER INSTRUCTIONS—SPURT3: Statistical Computations

STIX1, STIX2, STIX3

SUBROUTINE STIX1, STIX2, and STIX3 (TABLE, VMAX, VMIN, NINK, VAB, 
ITITLE, LNTHTL)

TYPE REAL VAB, TABLE, VMAX, VMIN

TYPE INTEGER NINK, LNTHTL

ITITLE is a Hollerith constant

These three subroutines accumulate and print out a frequency table and, optionally, plot a CalComp histogram. The calls to these STIX are

CALL STIX1(TABLE, VMAX, VMIN, NINK, VAB, ITITLE, LNTHTL)
CALL STIX2(TABLE, VMAX, VMIN, NINK, VAB, ITITLE, LNTHTL)
CALL STIX3(TABLE, VMAX, VMIN, NINK, VAB, ITITLE, LNTHTL)

VAB is the variable being accumulated; TABLE is the frequency table used for accumulating a count of the different values of the variable VAB; NINK is the number of increments desired in the table; VMAX and VMIN are the maximum and minimum values, respectively, in the table.

The dimension TABLE must be specified in the user's program and must have at least NINK + 4 entries. The additional four entries count those values of VAB that are either greater or less than the range specified by VMAX and VMIN (i.e., overflow and underflow) and also accumulate a running sum and sum of squares for use by STIX3.

ITITLE is a Hollerith constant (defined either in a DATA statement or in the calling sequence) that is used to label the plot produced by STIX. LNTHTL is the length of the title, specified as the number of characters in ITITLE.

NOTE: The user is required to pass the same parameters to STIX1, STIX2, and STIX3 for each user-defined frequency table being built and plotted.

The three routines are used as follows. STIX1 will initialize the frequency table to zero. STIX2 will add a one to the proper counter in TABLE associated with the current value of VAB. STIX3 will print out the frequency table TABLE and a plot of the histogram will occur at the user's option. If a CalComp plot has been made, the user is cautioned to place a call to ENDPLT before any possible program terminations. (See Ref. 1, p. 6.3.)

continued
STIX4

SUBROUTINE STIX4 (X,N,XMIN,XMAX,STADEV,XBAR)
    TYPE REAL X,XMIN,XMAX,STADEV,XBAR
    TYPE INTEGER N

STIX4 takes a one-dimensional array X(1), ..., X(N), which represents values of a variable X, and computes the mean (XBAR), standard deviation (STADEV), maximum value (XMAX), and the minimum value (XMIN). The call is of the form CALL STIX4 (X,N,XMIN, XMAX,STADEV,XBAR), where X is an array of length N that is defined in the calling program. XMIN, XMAX, STADEV, and XBAR are the values returned by the arguments of STIX4.

STIX5

SUBROUTINE STIX5 (X,Y,N,CORR)
    TYPE REAL X,Y,CORR
    TYPE INTEGER N

STIX5 requires two one-dimensional arrays, X(1), ..., X(N) and Y(1), ..., Y(N), which represent the values of two variables X and Y. If the correlation coefficient of these two arrays is desired, then the following call is made to STIX5: CALL STIX5 (X,Y,N,CORR). This call will return to the user in CORR the value of the correlation coefficient between the two arrays X and Y.

STIX6

SUBROUTINE STIX6 (A,N,AMEDIAN,RANGE)
    TYPE REAL A,AMEDIAN,RANGE
    TYPE INTEGER N

STIX6 is designed to rank a one-dimensional array A and return the value of the median and the range of the data contained in the N data entries of A. STIX6 is called by CALL STIX6 (A,N, AMEDIAN,RANGE). The routine uses a bubble sort to rank the data. This enables the user to call STIX6, with dummy parameters for AMEDIAN and RANGE, and use it just to sort a one-dimensional array.

NOTE: For ranking two-dimensional arrays, see List-Processing and Queue-Manipulation, SPURT4: subroutines RANKQS, IRNKQS, RANKQL, and IRNKQL.

STIX7

SUBROUTINE STIX7 (X,N,XI,XF,DX,MUNIT,ISTAT,MODE,IHIST,ITITLE,LNTH)
    TYPE REAL X,XI,XF,DX
TYPE INTEGER N, ISTAT, MODE, IHIST, LNTH 
MUNIT and ITITLE are Hollerith constants

The call for STIX7 is CALL STIX7 (X, N, XI, XF, DX, MUNIT, ISTAT, MODE, 
IHIST, ITITLE, LNTH). STIX7 will produce a statistical description 
of the data found in the first N locations of array X. MUNIT is 
a variable that contains the units in which the members of X are 
measured. ISTAT is a control parameter. If its value is nonzero, 
the sample size, mean, standard deviation, standard error, mini-
mum value, maximum value, and range will be printed out. If ISTAT 
equals zero, these will be suppressed.

In a similar manner, a nonzero value for MODE will cause the mode 
to be printed out and a nonzero value of IHIST will cause a histo-
gram to be printed on the line printer. ITITLE is the title of 
the histogram and consists of LNTH characters, where LNTH ≤ 90. 
If the mode and/or histogram is desired, XI, XF, and DX must be 
defined. (They must always be included in the calling sequence, 
however.) XI denotes the initial value for the histogram; XF is 
the final value; DX is the increment. STIX7 is set up to handle 
100 increments between XI and XF. The value printed for each DX 
is the benchmark (median) value of that interval.

WARNING: If ISTAT = MODE = IHIST = 0, no printout will result. 
Make sure that at least one of these parameters is nonzero.

Job Deck Structure

Job Cards
Run(s)
LIBRARY(SPURT2, SPURT3, SPURT6)
LGO.
7–8–9 (EOR Card)
Main-Problem Card(s)
PROGRAM(INPUT, OUTPUT, PUNCH, PLOT, TAPE5=INPUT, 1 TAPE6=OUTPUT, 
TAPE7=PUNCH, TAPE17=PLOT)
.
.
END
Subroutine Card(s)
SUBROUTINE ACTION1
.
.
END
SURROUTINE ACTIONZ
.
.
END
7–8–9 (EOR Card)
Data Cards
6–7–8–9 (EOI Card)

NOTE: The file assignments on the Main-Program header card are the standard SPURT assignments.
SAMPLE INPUT

SUBROUTINE TESTCOR
      DO 10 W1 = 1, 2000
         X(I) = RANF(-1)
      Y(I) = RANF(-1)
      PRINT 90
  90 FORMAT(5/01 TEST OF SPURT5 AND SPURT7)
      DO 10 Y1 = 1, 2000
         Y(I) = RANF(-1)
      PRINT 20
  20 FORMAT(3/0' COEFFICIENT OF CORRELATION BETWEEN *AT1
         *AT2, AND *AT3, * = *F10.4)
      CALL STICKS(X, 2000, K)
      PRINT 20 N1, N2
      CALL STIRY(X, 2000, AMEDIAN, RANGE)
      PRINT 30 N1, AMEDIAN, RANGE
      RETURN
  30 FORMAT(/P FOR THE VARIABLE *AT1 MEDIAN = *5F10.4, RANGE = *F10.4)
      END
SAMPLE OUTPUT

***** STIX4 *****  DATE 01/27/70  PAGE NO. 23

TEST OF STIX4

MIN = 14.065
MAX = 18.956
SIGMA = 1.6445
MEAN = 16.631
** CP TIME SINCE LAST CALL **  .024 SECONDS

TEST OF STIX4 AND STIX6

COEFFICIENT OF CORRELATION BETWEEN X AND X = 1.00000
COEFFICIENT OF CORRELATION BETWEEN X AND Y = -.92223
** CP TIME SINCE LAST CALL **  .129 SECONDS

FOR THE VARIABLE X  MEDIAN = .4946  RANGE = .9996
** CP TIME SINCE LAST CALL **  1.303 SECONDS

FOR THE VARIABLE Y  MEDIAN = .4809  RANGE = .9979
** CP TIME SINCE LAST CALL **  1.273 SECONDS

188
USER INSTRUCTIONS—SPURT4: Analog Simulators

The following two routines are designed to simulate an analog computer on a digital computer, thereby making it possible to obtain effects similar to those displayed by a hybrid. Both routines operate as integrators in an analog computer, integrating negatively with respect to time. Input (XIN) must be a floating-point number. Output (YOUT and TOUT) is two arrays, one giving values of time and the other giving the corresponding negative integral values. The output arrays are not printed out. Initial conditions Y may also be specified.

ANALOG

SUBROUTINE ANALOG(XIN,Y,YOUT,TOUT,T,DELTA,LIMIT,N)
TYPE REAL XIN,Y,YOUT,TOUT,T,DELTA
TYPE INTEGER LIMIT,N

Subroutine ANALOG is called in the following form: CALL ANALOG(XIN, Y,YOUT,TOUT,T,DELTA,LIMIT,N). The input value is XIN. YOUT and TOUT are the output arrays of integrated value and time, respectively. The user is cautioned to dimension YOUT and TOUT to the expected size in his program. Integration is carried out at discrete time intervals DELTA and for the number of intervals LIMIT. Any number of successive integrations N can be carried out. Thus, for \( N = 1 \), a first-order integration is performed; for \( N = 2 \), a second-order integration is performed; etc. Note that LIMIT and N are fixed-point numbers. T is the initial value of time, and TOUT(1) = T. Thus, TOUT(2) = T + DELTA and the last value TOUT(LIMIT) = T + DELTA * (LIMIT -1). Y is a vector of initial conditions of size N, where Y(1) is the initial value for the first-order integration, Y(2) for the second-order, etc. Thus, Y(1) gives the initial value of y; the output, Y(2), gives the initial value of \( \int y\, dt \), etc. The array Y(N) must be dimensioned and provided in the user's program.

A second-order integration works exactly as do two successive integrators on an analog computer. Thus, for a positive input, the output array will also be positive.

SECND

SUBROUTINE SECND(XIN,YIC,DAMP,FREQ,T,YOUT,TOUT,DELTA,LIMIT)
TYPE REAL XIN,YIC,DAMP,FREQ,T,YOUT,TOUT,DELTA
TYPE INTEGER LIMIT

This routine provides the solution to the commonly used control equation
F(x) = \frac{d^2y}{dt^2} = 2zw_n \frac{dy}{dt} + w_n^2 y,

where z is the damping coefficient, w_n is the natural frequency, and F(x) is a constant forcing function (such as a step input). SECND is equivalent to the analog connection

and is similar in operation to the subroutine ANALOG.

The form of the call is CALL SECND(XIN,YIC,DAMP,FREQ,T,YOUT,TOUT, DELTA,LIMIT), where T is the initial value of time, DELTA is the integration interval, LIMIT the number of values calculated, and YOUT and TOUT the output arrays. All are defined as in subroutine ANALOG. Only LIMIT is fixed-point and YOUT and TOUT must be dimensioned in the user's program. YIC is the initial value of YOUT. Thus, YOUT(1) = YIC. XIN is the forcing function F(x). DAMP is the damping coefficient z. FREQ is the natural frequency w_n.

Job Deck Structure

Job Cards
LIBRARY(SPURTS,SPURT6)
LGO.
7-8-9 (EOR Card)
Main-Problem Card(s)
PROGRAM(INPUT,OUTPUT,PUNCH,PLOT,TAPE5=INPUT, 1 TAPE6=OUTPUT,
TAPE7=PUNCH,TAPE17=PLOT)
.
.
END
Subroutine Card(s)
SUBROUTINE ACTION1
.
.
END
SUBROUTINE ACTIONZ

END

7-8-9 (EOR Card)
Data Cards
6-7-8-9 (EOI Card)

NOTE: The file assignments shown on the Main-Program header card are the standard SPURT assignments.
SAMPLE INPUT/OUTPUT

The input for the Analog Simulators (SPURT4) will consist of floating-point numbers that are integrated negatively with respect to time. The output will be two arrays, one giving values of time and the other giving the corresponding negative integral values.
USER INSTRUCTIONS—SPURT5: List-Processing and Queue-Manipulation

ADFIFO, ADLIFO, REMOVE, PURGE, DISPL, IDISPL, NUMLST

SUBROUTINE ADFIFO(LIST,N,M,ENTRY)
SUBROUTINE ADLIFO(LIST,N,M,ENTRY)
SUBROUTINE REMOVE(LIST,N,M,ENTRY)
SUBROUTINE PURGE(LIST,N,M)
SUBROUTINE DISPL(LIST,N,M)
SUBROUTINE IDISPL(LIST,N,M)
INTEGER FUNCTION NUMLST(LIST)
INTEGER N,M
REAL (or INTEGER) LIST, ENTRY

The manipulations of queues may be handled by the four subroutines ADFIFO, ADLIFO, REMOVE, and PURGE. REMOVE will remove the top (or first) element from the list. ADLIFO inserts an element at the top of the list, which will be removed before any of the elements presently on the list. ADFIFO adds an entry at the bottom of the list, which will be removed only after all the elements presently on the list are gone. Lists may be manipulated by using both ADLIFO and ADFIFO on the same list array. A printout of the contents of a list may be achieved by use of either DISPL or IDISPL.

The general form of the calls to these routines is as follows.

CALL ADFIFO(LIST,N,M,ENTRY)
CALL ADLIFO(LIST,N,M,ENTRY)
CALL REMOVE(LIST,N,M,ENTRY)
CALL PURGE(LIST,N,M)
CALL DISPL(LIST,N,M)
CALL IDISPL(LIST,N,M)
NUM = NUMLST(LIST)

In the calling sequence, LIST is the user-defined array of size NXM and ENTRY is the user-defined vector, dimensioned (1,M), used as a buffer. As LIST(1,1) contains the list pointer, only N - 1 entries (each of size 1X M) are allowed in any list.

NOTE: Before loading the initial entry to any list, be sure to clear the word LIST(1,1) to zero.

The subroutines are designed to place the contents of a buffer (ENTRY) onto a user-defined list. The buffer ENTRY should be filled by the user's program before ADFIFO or ADLIFO are called. In the continued
case of REMOVE, the first entry is taken from the list and placed in the buffer ENTRY. The user may then use its contents.

The integer function NUMLST will return the number of entries in a given list. IQ = NUMLST(LIST3) will assign a value to IQ equal to the number of entries in LIST3.

RANKQS, IRNKQS, RANKQL, IRNKQL

SUBROUTINE RANKQS(LIST,N,M,J)
SUBROUTINE IRNKQS(LIST,N,M,J)
SUBROUTINE RANKQL(LIST,N,M,J)
SUBROUTINE IRNKQL(LIST,N,M,J)

INTEGER N,M,J
REAL or INTEGER LIST

Four routines are provided to rank queues formed by other SPURT list-processing routines. They may be used by coding of the following form.

CALL RANKQS(LIST,N,M,J)
CALL IRNKQS(LIST,N,M,J)
CALL RANKQL(LIST,N,M,J)
CALL IRNKQL(LIST,N,M,J)

where LIST is the user-defined array of size NXM and J < M represents the "entry" position that is to be used to produce the desired ranking.

Subroutine RANKQS ranks floating-point numbers with the smallest first and IRNKQL ranks integer numbers with the largest first. Similarly, IRNKQS ranks integer numbers in ascending order.

SEARCH, INSERT, DELETE

INTEGER FUNCTION SEARCH(LIST,N,M,J,ID)
SUBROUTINE INSERT(LIST,N,M,ENTRY,INDEX)
SUBROUTINE DELETE(LIST,N,M,ENTRY,INDEX)
INTEGER N,M,J,INDEX
REAL or INTEGER LIST,ENTRY,ID

Three routines are provided to facilitate the locating, deleting, and inserting of entries in the middle of a list. The use of the INTEGER SEARCH function in the form INDEX = SEARCH(LIST,N,M,J,ID), where SEARCH is of type INTEGER, will search LIST (of size NXM) in entry position J for the desired ID. If a match is found, SEARCH returns the position in the list of the first entry to contain the desired ID. If no match is found, SEARCH returns a zero value.
Thus, SEARCH may be used within an arithmetic or logical IF statement to determine the presence or absence of a desired entry. The user may use either an integer or a real variable as the fifth argument in a call to SEARCH.

If the ith entry has been located (say, by using SEARCH), then it may be removed by a call to DELETE in the form of CALL DELETE(LIST, N,M,ENTRY,INDEX), where INDEX is the number of the desired entry to be removed from LIST and placed in ENTRY.

To place a new ith entry in the list, a call to INSERT in the form of CALL INSERT(LIST,N,M,ENTRY,INDEX) will create a new ith entry (i is noted by INDEX).

Job Deck Structure

Job Cards
Run(s)
LIBRARY(SPURT4,SPURT6)
LGO.
7-8-9 (EOR Card)
Main-Problem Card(s)
  PROGRAM(INPUT,OUTPUT,PUNCH,PLOT,TAPE5=INPUT, 1 TAPE6=OUTPUT,
   TAPE7=PUNCH,TAPE17=PLOT)
  ...
  END
Subroutine Card(s)
  SUBROUTINE ACTION1
  ...
  END
  ...
  ...
  SUBROUTINE ACTIONZ
  ...
  END
7-8-9 (EOR Card)
Data Cards
6-7-8-9 (EOI Card)
SAMPLE INPUT

SUBROUTINE LISTI
COMMON A1,B1,C1,A+B+C
DIMENSION A(13,13), B(12,17), C(119,7)
DIMENSION A(20,20), R(10,10), C(20,10), NMROWS(20), NMCOLS(20)
DIMENSION ENTRY(110)
DATA NMROWS, NMCOLS, ENTRY, INMCOL, INMROWS, NMCOLS, NMROW
INTEGER AOIC, A111, C1
DIMENSION A111(3,3), A1(2620), R(10,20), C(20,10), NMROWS(204), NMCOLS(265)
DIMENSION IENTRY(10)
DATA INMROWS, INMCOL, INMROWS, NMCOLS, NMROW, NMCOL

I=1,10
DO 30 J=1,10
K=J
 ENTRY(I,J) = ENTRY(I,J) + 1
CALL INSOUT (ENTRY,I,J,NMROWS,NMCOLS)
CALL IDISPL (C,20,10)
DUMETIMELP1 I DO 10 I=1,10
CONTINUE
RETURN
END
**LIST DISPLAY ROUTINE INITIATED**

**OUTPUT IN FIXED POINT FORMAT**

<table>
<thead>
<tr>
<th>ENTRY NUMBER</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**END OF OUTPUT BY IDISPL**

---

**CP TIME SINCE LAST CALL**  **.042 SECONDS**

<table>
<thead>
<tr>
<th><strong>R1</strong></th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>C1</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C2</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C3</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C4</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C5</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C6</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C7</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C8</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C9</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C16</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**List Display Routine Initiated**

**Output in Fixed Point Format**

<table>
<thead>
<tr>
<th>Entry Number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Entry Number**

<table>
<thead>
<tr>
<th>Entry Number</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**End of Output by IOISPL**

**CP Time Since Last Call**

- 0.053 seconds

**Insout**

<table>
<thead>
<tr>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
<th>C7</th>
<th>C8</th>
<th>C9</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>101</td>
<td>102</td>
<td>103</td>
<td>104</td>
<td>105</td>
<td>106</td>
<td>107</td>
<td>108</td>
</tr>
</tbody>
</table>
LIST DISPLAY ROUTINE INITIATED...

OUTPUT IN FIXED POINT FORMAT

<table>
<thead>
<tr>
<th>ENTRY NUMBER</th>
<th>1</th>
<th>102</th>
<th>103</th>
<th>104</th>
<th>105</th>
<th>106</th>
<th>107</th>
<th>108</th>
<th>109</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>110</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ENTRY NUMBER</th>
<th>2</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ENTRY NUMBER</th>
<th>3</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

END OF OUTPUT BY IDISPL

**CP TIME SINCE LAST CALL**  0.062 SECONDS
LIST DISPLAY ROUTINE INITIATED

DATE 01/27/78

OUTPUT IN FIXED POINT FORMAT

ENTRY NUMBER
1  201  210  202
2  203  102  101
3  110  12  13
4  30  22  21
5  30  22  21

PAGE NO. 39

END OF OUTPUT BY IDISOL

EDUCATIONAL INFORMATION NETWORK
000 0071
SPURTS

continued
**DISPLAY ROUTINE INITIATED**

**OUTPUT IN FIXED POINT FORMAT**

<table>
<thead>
<tr>
<th>ENTRY NUMBER</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ENTRY NUMBER</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**END OF OUTPUT BY IDISPL**

**CP TIME SINCE LAST CALL**

0053 SECONDS

---

**INSCUT**

<table>
<thead>
<tr>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
<th>C7</th>
<th>C8</th>
<th>C9</th>
<th>C10</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>26</td>
</tr>
</tbody>
</table>

---

**Continued**
LIST DISPLAY ROUTINE INITIATED
ENTRY NUMBER 21
DATE 01/27/70
PAGE No. 45

OUTPUT IN FIXED POINT FORMAT
END OF OUTPUT

NORTHERN UNIVERSITY SYSTEM

DATE 01/27/70
PAGE No. 46

END OF OUTPUT BY IDISPL

EDUCATIONAL INFORMATION NETWORK
000 0071
SPURTS

continued
LIST DISPLAY ROUTINE INITIATED...

OUTPUT IN FIXED POINT FORMAT

NO ENTRIES ON LIST

END OF OUTPUT BY IDISPL

.. CPU TIME SINCE LAST CALL .. 0.40 SECONDS
USER INSTRUCTIONS—SPURT6: Matrix and Graphical Output

OUT, IOUT

SUBROUTINE OUT(C,N,NAME)
SUBROUTINE IOUT(K,N,NAME)

TYPE REAL C
TYPE INTEGER N,K

Name is vector of Hollerith constants

Subroutine OUT prints out a square matrix with row and column headings. The form of the calling statement is CALL OUT(C,N,NAME), where C is the name of a two-dimensional square array, N is the number of rows (and columns) of the matrix C, and NAME is the name of a vector, dimensioned N in the user's program that contains the names to be printed over the columns (and along the rows). The heading of row j and column j will be the same, for j = 1, ..., N. The vector NAME must contain Hollerith constants of less than or equal to M (installation defined) characters defined in a DATA or assignment statement.

The matrix is printed out in sets of N rows by 10 columns. When the matrix printouts are placed beside one another, the entire matrix is represented.

The call to OUT assumes that floating-point numbers are to be printed out in E12.4 format. If the user wishes to print out fixed-point numbers, he should CALL IOUT(K,N,NAME), where K is an array of TYPE INTEGER and is to be printed out in I format.

NSOUT, INSOUT

SUBROUTINE NSOUT(C,NH,NV,NAMEH,NAMEV)
SUBROUTINE INSOUT(K,NH,NV,NAMEH,NAMEV)

TYPE REAL C
TYPE INTEGER NH,NV,K

NAMEH and NAMEV are vectors of Hollerith constants

Subroutine NSOUT is similar to subroutine OUT, except that NSOUT causes a nonsquare matrix, with appropriate row and column headings, to be printed out. The form of the CALL statement is CALL NSOUT(C, NH,NAMAH,NAMEV), where C is a two-dimensional array, NH is the number of rows of the array C, and NV is the number of columns of C. NAMEH is the name of a vector, dimensioned NH in the calling programs, that contains the names of the row headings. NAMEV is the name of a vector, dimensioned NV in the calling program, that contains the names of the column headings.

Calling the NSOUT subroutine causes the matrix C to be printed out continued
The subroutine differs from OUT in that it allows column headings to differ from row headings.

The call to NOUT assumes that floating-point numbers are to be printed out in E12.4 format. If the user wishes to print out fixed-point numbers, he should CALL INSOUT(K,NH,NAMEH,NAMEV), where K is an integer array to be printed out in I format.

GRAPH

SUBROUTINE GRAPH(NAMEX,LNTHX,NAMEY,LNTHY,X,Y,N)
TYPE REAL X,Y
TYPE INTEGER LNTHX,LNTHY,N
NAMEX and NAMEY are Hollerith constants

GRAPH produces two-dimensional graphs or plots on the CalComp plotter, requiring two one-dimensional arrays A and B of the same size, where A = a1, ..., an, B = b1, ..., bn, and yielding a plot whose points are connected by straight line segments. When the user desires to print out a graph, he uses the following statement: CALL GRAPH(NAMEX,LNTHX,NAMEY,LNTHY,X,Y,N), where NAMEX is the label for the X axis. LNTHX is the number of characters in the title NAMEX. Similarly, NAMEY is the label for the Y axis and LNTHY is the number of characters in it. X and Y are the names of one-dimensional arrays (of the same size) to be plotted. N is the total number of points to be plotted, plus two. That is, if each array contains p entries, then N would be p + 2.

The following statement must appear immediately before all legal termination points in the user's program: CALL ENDPILT. This finalizes the output for the CalComp plotter. Note that no parameters are to be passed within the call to NEDPILT.

NOTE: The user may have titles of a length greater than M (installation defined) only if they are defined as Hollerith characters in the calling sequence to the GRAPH subroutine.

PAGE

SUBROUTINE PAGE(NAME)
NAME is a Hollerith constant

This routine is called to produce a page ejection in the printout of a program. It will inset up to M (installation defined) characters, found in NAME, into a heading that includes the date and page number. The most obvious use of NAME is to supply the name of the subroutine calling PAGE. For example, CALL PAGE(NAME). The routines in SPURT that produce page ejections (CLOCK, STIX2, etc.) all do so by calling
PAGE. Thus, if a page number is to be meaningful to the user, he also must eject pages by a call to PAGE.

Job Deck Structure

Job Cards
LIBRARY(SPURT3,SPURT6)
LGO.
7-8-9 (EOR Card)
Main-Problem Card(s)
PROGRAM(INPUT,OUTPUT,PUNCH,PLOT,TAPE5=INPUT, 1 TAPE6=OUTPUT, TAPE7=PUNCH,TAPE17=PLOT)
.
.
END
Subroutine Card(s)
SUBROUTINE ACTION1
.
.
END
.
.
SUBROUTINE ACTION2
.
.
END
7-8-9 (EOR Card)
Data Cards
6-7-8-9 (EOI Card)

NOTE: The file assignments shown on the Main-Program header card are the standard SPURT assignments.
SUBROUTINE example

*********This program is part of the Northwestern University SPURT

A simulation package for university research and teaching

****************************************

COMMON/GPLOT/UNUSED(5),IPLTSW
DIMENSION T(104),TYPE REAL NEEXP
CALL STIX1(T=20.0+0.0*100.0.VALUE=22HDEPTH OF WATER IN TANK:22)
CALL STIX2(T=20.0+0.0*100.0.VALUE=22HDEPTH OF WATER IN TANK:22)
RETURN

eng

continued
## Depth of Water in Tank

<table>
<thead>
<tr>
<th>Upper Limit</th>
<th>Observed Frequency</th>
<th>Per Cent of Total</th>
<th>Cumulative Percentage</th>
<th>Cumulative Remainder</th>
<th>Multiple of Mean Deviation</th>
<th>Cumulative Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNDERFLOW</td>
<td>0.000</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>-2.892</td>
</tr>
<tr>
<td>±200</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.014</td>
<td>-2.893</td>
</tr>
<tr>
<td>±400</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.027</td>
<td>-2.874</td>
</tr>
<tr>
<td>±600</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.041</td>
<td>-2.774</td>
</tr>
<tr>
<td>1.000</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.068</td>
<td>-2.463</td>
</tr>
<tr>
<td>1.200</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.082</td>
<td>-2.464</td>
</tr>
<tr>
<td>1.400</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.095</td>
<td>-2.466</td>
</tr>
<tr>
<td>1.600</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.109</td>
<td>-2.457</td>
</tr>
<tr>
<td>1.800</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.123</td>
<td>-2.437</td>
</tr>
<tr>
<td>2.000</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.000</td>
<td>0.136</td>
<td>-2.408</td>
</tr>
<tr>
<td>2.200</td>
<td>0.333</td>
<td>0.333</td>
<td>0.333</td>
<td>0.333</td>
<td>0.150</td>
<td>-2.399</td>
</tr>
</tbody>
</table>

This data is accompanied by graph number 10.
<table>
<thead>
<tr>
<th>GRAPH NUMBER</th>
<th>10 HAS JUST BEEN PLOTTED ON CALCOMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>THE PLOT IS OF</td>
<td><strong>DEPT</strong>H OF WATER IN TANK</td>
</tr>
<tr>
<td>VS.</td>
<td><strong>FREQUENCY (PER CENT)</strong></td>
</tr>
<tr>
<td>END OF OUTPUT BY GRAPH</td>
<td></td>
</tr>
</tbody>
</table>
COST ESTIMATE

For the SPURT TESGEN package listed on the Sample Input, the total running time was 91.59 seconds for the central-processor time and 46.597 seconds for the peripheral-processor time. Chargeable computer time was $14.16

Charge to user = computer time + postage and handling + network overhead

= $14.16 + $15.00 + network overhead

= $29.16 + network overhead

CONTENTS—SPURT

pages

1– 4 Abstract & Identification
User Instructions

5– 7 SPURT
9–13 SPURT1
21–25 SPURT2
35–38 SPURT3
41–43 SPURT4
47–49 SPURT5
61–63 SPURT6

I/O

15–19 SPURT1
27–33 SPURT2
39–40 SPURT3
45 SPURT4
51–59 SPURT5
65–67 SPURT6

69 Cost—Contents

69
FORTRAN Program to Assist in the Process of Political Reapportionment

18 FSU BELOW

Computer Center
The Florida State University

William Below, Consultant
Assembly Committee on Elections and Reapportionment
California Legislature

CDC FORTRAN IV

CDC 6400-65K

Decks and listings presently available

Raymond Soller, Program Librarian/EIN Techrnical Representative, Computing Center, Florida State University, Tallahassee, Fla. 32306
Tel.: (904) 599-4770

BELOW is a program to assist with the development of political reapportionment plans using a set of procedures representative of the methods employed by legislatures to reapportion themselves. The actual algorithms utilized by the program are modifications of the computational techniques originally outlined by Stuart Nagel.1

The programmed procedures are those which follow. An initial district plan is determined by assigning geographic units to districts. This initial plan is fed into the processor along with demographic and political data for each unit. The program then tests each possible move of a unit from one district to another and then each trade of units between adjacent districts against a set of criteria. If the plan is approved according to the criterion; then the move or trade is made permanent. Where no more useful moves or trades can be made, the results are printed out and the program is terminated.

continued
The criterion used by the program is expressed as a linear combination of three quantities, SVEA, SVCA, SVPA;

\[ \text{CRIT} = \text{WE} \cdot \text{SVEA} + \text{WC} \cdot \text{SVCA} + \text{SVPA} \]

where SVEA, SVCA, and SVPA are measures of population equality, compactness, and conformance to political goals, and WE and WC are weighting coefficients for the purpose of establishing the relative importance of the three quantities. CRIT, the criterion, is recomputed for each tentative move or trade. If its value is less than its previous value, the move or trade is consummated.

Population Equality

The measure of population equality is given by

\[ \text{SVEA} = \sum_{J=1}^{\text{NDA}} \left( \frac{\text{PD}(J) - \text{AVGPOP}(J)}{\text{AVGPOP}(J)} \right)^2 \cdot 100 \]

where NDA is the number of districts in the area, PD(J) is the population of district J and AVGPOP(J) is the ideal population. SVEA may be described as the sum of the squared fractional deviations from the ideal population for all districts in the area.

Multiplication of the fractional deviation by 100 is not significant to the operation of the program; it simply indicates that the deviation is output as a percentage. The use of the fractional rather than the absolute deviation is significant because it makes the effect of WE, (the relative importance of population equality), independent of the ideal population. Similar considerations apply to the decision to use total rather than average deviation for the area. If average deviation were used, then the proper setting of WE would depend on the number of districts in the area.

The opportunity for the user to set the ideal population for each district independently may at first seem arbitrary. In practice, this allows the program to be used on areas which contain portions of districts as well as whole districts.

This method of measuring population equality does not differ markedly from methods used by others in the field. One obvious alternative, namely, use of the sum of the absolute deviations, might be quite acceptable as a measure of equality to a student government, but it suffers from some computational disadvantages.

continued
As a population unit moves from one district to another, the absolute deviation changes only if one district is above, while the other is below the ideal population. The formula given here for SVEA however, has the convenient property that when two districts are both above or below the ideal population, a unit moved from one district to the other causes a change in SVEA, proportional to the difference in population between the two districts.

Compactness

The method originally tried for measuring compactness determines the "population moment" around the population center of the district. This method is attractive because it tends to minimize travel distance within the district and keep concentrations of population unbroken by district lines.

If there is a large number of units per district, the population moment method, unfortunately, can yield districts which are numerically compact, yet have very irregular boundaries. Since a legislature, as it reapportions itself, prefers to avoid even the appearance of gerrymandering, the author found the population moment method unsatisfactory for his purposes.

The alternative method, now in use, involves estimation of the perimeter of each district. The estimation is inherently inexact, but tends to produce district lines which are politically acceptable.

The program counts the perimeter elements in the perimeter of each district. A perimeter element is that portion of a unit perimeter which is shared with one other unit. Of course, these elements vary in length, but the program assumes each to have unit length.

One of the attractive features of the perimeter method is that it dispenses with the use of the X and Y coordinates required for the population moment method. What may be unattractive to some users is that the perimeter method does not tend to improve or even affect the overall shape of a district unless the initial district plan deviates widely from the desired characteristics.

Thus, if the program is given a district which is rectangular, and which is much longer than it is wide, the lines will be kept straight, but there will be little or no tendency for that district to change toward a square or circular shape. This property is often quite convenient in the legislative environment.

continued
In principle, it would be possible to measure the length of each perimeter element, include the data in the information fed to the processor, and work with a true perimeter measurement. Making those measurements, however, would be even more tedious than preparing the X and Y data, and may not necessarily improve program performance.

Political Considerations

The political portion of the criterion, CRIT, is given by

\[
SVPA = \sum_{J=1}^{NDA} WP(J) \left( \frac{PAD(J) \cdot 100}{PBD(J)} - DESPR(J) \right)^2
\]

where WP (J) is a political weighing coefficient that may be set independently for each district, PAD (J) and PBD (J) are political quantities for district and DESPR (J) is the desired ratio of PAD (J) to PBD (J) expressed as a percentage.

PAD (J) and PDB (J) may stand for different things in different districts according to a number called MODE (J). If MODE (J) were set to 1, for example, and DESPR (J) to 55, then the program would set PAD (J) to equal registered Democrats and PBD (J) to equal total registered voters, and the goal for Democratic registration in that district would be 55%.

In the same manner, setting MODE (J) to 2 or 3 will establish a goal for the percentage of Negroes or persons with a Spanish surname in the population. The information for each unit necessary to establish any of the three proportions is carried in the memory of the processor. Consequently, there are no restrictions on moving a unit between districts with different modes.

Constraints on Moving and Trading

By the inclusion of the one card per unit in the data deck, the user may forbid the program to move or trade any number of units out of their original districts. The most common use of this feature is to ensure that incumbents are left in their own districts. Another use is to protect the integrity of municipal areas.

REFERENCES


2. Below, W., "The Computer as an Aid to Legislative Reapportionment," (Report to California Legislature, Assembly Committee on Elections and Reapportionment).
Unit Card

One Unit Card is needed for each Census tract, enumeration district, precinct, or other geographical unit.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Identification</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>1: Unit Card label</td>
</tr>
<tr>
<td>2-7</td>
<td>UNIT CODE</td>
<td>alphanumeric unit code used to identify a unique geographic unit.</td>
</tr>
<tr>
<td>8-11</td>
<td>PU</td>
<td>population count within the unit</td>
</tr>
<tr>
<td>12-15</td>
<td>P1</td>
<td>population characteristic one</td>
</tr>
<tr>
<td>16-19</td>
<td>P2</td>
<td>population characteristic two</td>
</tr>
<tr>
<td>20-23</td>
<td>P3</td>
<td>population characteristic three</td>
</tr>
<tr>
<td>24-27</td>
<td>P4</td>
<td>population characteristic four</td>
</tr>
<tr>
<td>38-79</td>
<td>TOUCHLIST</td>
<td>listing of all unit codes which are adjacent to the unit identified in Cols. 2-7.</td>
</tr>
<tr>
<td></td>
<td>partitioned into 7 fields of 6 columns</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>NL</td>
<td>number of neighbors entered into TOUCHLIST field</td>
</tr>
</tbody>
</table>

In a single pass of the program the selected data that is tabulated into district totals (P1/P2, P3/POPULATION, P4/POPULATION) is determined by the value of MODE.

The TOUCHLIST is used for establishing and maintaining contiguity, so all rules and assumptions concerning contiguity should be considered when compiling the TOUCHLIST. The TOUCHLIST is also used to measure compactness. The computations are more precise if reciprocity is observed. Reciprocity implies that if Unit A is a member of the TOUCHLIST for Unit B, then Unit B is also a member of the TOUCHLIST for Unit A.

TOUCHLIST Continuation Card

If there are more than seven neighbors in a unit's TOUCHLIST the remaining neighbors may be inserted on a Continuation Card. A unit may have as many Continuation Cards as needed.

continued
Columns | Identification | Contents
---|---|---
1 | | 2: Unit Continuation Card label
38-79 | TOUCHLIST | listing of unit code neighbors continued from previous card
80 | NL | number of neighbors entered into the TOUCHLIST field

The total number of entries for the complete TOUCHLIST must not exceed 8500.

"Eight" Card
This card signals the end of the initial district data.

Columns | Identification | Contents
---|---|---
1 | | 8: "Eight" Card label

Status Card
A Status Card is necessary if the possible moves or trades of a unit out of its district are to be restricted. One card is required for each such specified unit, and must follow an "Eight" Card.

Columns | Identification | Contents
---|---|---
1 | | 4: Status Card label
2-7 | UNIT CODE | alphanumeric unit code used to identify a unique geographic unit.
11 | | 1: flag which restricts unit movements

"Nine" Card
This card initiates the district processing, and must follow the status cards if present or an "Eight" Card.

Columns | Identification | Contents
---|---|---
1 | | 9: "Nine" Card Label

Parameter Update Card
An update pass is possible by changing the label field (Col. 1) to a 5 and by adjusting the proper parameters. All entries must be present even if they repeat the items on the Parameter Card. If it is desired to end the restrictions on a unit, then include a Status Card with a zero in Col. 11.

continued
Any number of reruns can be made, but each successive run will continually distort its relation to the initial district plan.

Sense Switches

Sense switches can be set from the console by the machine operator or by a SWITCH CARD in the set of system control cards accompanying the job.

SWITCH, n.

where n is a switch from 1 to 6, which should be set on.

- sense switch 2 on - suppress most printer output.
- sense switch 3 on - (by operator intervention only) Terminates program in orderly manner.
- sense switch 5 on - suppress trading mode of operation which normally begins when no more moves can be made

Output

The printed output of the program consists of a one page report for each district and an area summary for the initial plan, for the final plan, and for each successive rerun.

Each district report includes a complete list of the units in the district, a partial list showing only units on the edge of the district (for map drawing), and a numerical description of the district. The quantities "A" and "B" correspond to two of the quantities P1, P2, P3, or P4 according to the mode setting.

One line is printed summarizing the results of each major program loop. These lines appear between the initial and final district report pages. The numbers under NTM and NTT indicate the number of proposed moves and trades in each loop which satisfy the contiguity requirements and hence the number of times the program sections which tentatively recompute the criterion function are entered. These loop summaries are also printed on-line unless sense switch 2 is down.

Error Message

An error condition caused by improper data card punching results in an error message printed both off-line and on-line followed by exit. The message is head by "DATA ERROR" and consists of the following items.

continued
Item | Contents
--- | ---
IRD (1) | Col. 1 of last card read
IRD (2) | Cols. 2-7 of last card read
I | Internal unit counter
J | Internal district counter
K | Internal TOUCHLIST counter
M | See program listing
LPTYP | See program listing
JRET | See program listing

"Not in Area" Message

Each TOUCHLIST entry which does not have a corresponding Unit Card is listed under the heading "ON TOUCHLIST, NOT IN AREA." Two hundred such entries can be accommodated by the program. The presence of unmatched TOUCHLIST entries does not constitute an error condition and is common when a subset of the Unit Cards for an area is used for a program run.

"Not Contiguous" Message

A check for contiguity is made on the initial district plan. If any initial district is not contiguous a message of the form "DISTRICT J NOT CONTIGUOUS" is printed. The program does not stop since this condition is not necessarily disastrous to the program run. The operator may, however, be instructed to terminate execution when this message appears on-line.
SAMPLE INPUT

```
  0 100 75 600 10 -0 -0 -2L4. CONGESSIONAL REAPPORITIONMENT - BELOW -
  1 7071764 4601011 1 -0 -0 6 0 2 3 636 1296 3
  1 7650 407 4801012 1 -0 -0 1 2 4 636 636 5 4 2 6
  1 4511114 7001013 1 -0 -0 1 636 636 5 4 2 6
  1 6624672 7601021 1 -0 -0 75 63 8 9 4 636 636
  2 -0 -0 -0 -0 -0 -0 -0 -0 -0 -0 690 661 3
  1 4571171 4701031 1 -0 -0 7 0 1 1297 4
  1 7611129 5761032 1 -0 -0 7 0 6 1297 5
  1 8741176 4701037 1 -0 -0 10 65 5 9 7 5
  1 5771154 6741034 1 -0 -0 9 5 4 6 0 3 6
  1 5111140 7214112 1 -0 -0 38 40 44 36 34 23 1294
  1 7619446 9041117 1 -0 -0 39 45 42 37 4
  1 7620176 7217121 1 -0 -0 39 40 35 1296 4
  1 9679467 2351132 1 -0 -0 38 40 41 1293 141 142 1294
  1 40 65171116611133 1 -0 -0 39 41 44 39 38 5
  1 4116128 112111134 1 -0 -0 34 120 123 42 42 5
  1 42 6417147 7214115 1 -0 -0 36 47 45 48 4
  1 47 86510210211152 1 -0 -0 36 44 45 48 4
  1 47 7215279 4781153 1 -0 -0 40 44 126 43 35 6
  1 46 6611275 6681154 1 -0 -0 46 124 123 122 121 120 487
  2 -0 -0 -0 -0 -0 -0 -0 -0 -0 -0 42 43

```

### SAMPLE OUTPUT

**DISTRICT OUTPUT**

**UNIT N Cl IN TION.**

<table>
<thead>
<tr>
<th>DISTRICT</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>L.A. CONCESSIONAL RAPPORTMENT</td>
<td>- BELOW</td>
</tr>
</tbody>
</table>

#### UNITS

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

#### UNITS ON EDGE OF DISTRICT

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

#### POPULATION VARIATION

<table>
<thead>
<tr>
<th>VARIATION</th>
<th>PCT VARIATION</th>
<th>SQRD VARIATION</th>
<th>WEIGHTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>56625.0000</td>
<td>6465.0000</td>
<td>15.6878</td>
<td>246.1000</td>
</tr>
</tbody>
</table>

#### PROPORTION SQRD VARIATION

<table>
<thead>
<tr>
<th>4</th>
<th>9</th>
<th>6</th>
<th>9</th>
<th>6</th>
<th>9</th>
<th>6</th>
<th>9</th>
</tr>
</thead>
</table>

#### NO OF UNITS

<table>
<thead>
<tr>
<th>WP</th>
<th>DESPR</th>
<th>COMPACTNESS</th>
<th>WEIGHTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>99.0000</td>
<td>0.0100</td>
<td>100.0000</td>
<td>98.0000</td>
</tr>
</tbody>
</table>

**TOTAL CONTRIBUTION TO CRITERION = 703.1874**

continued
## L.R. CONGRESSIONAL REAPPORTIONMENT - BELOW

### UNITS

<table>
<thead>
<tr>
<th>No</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
<td>40</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>43</td>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
<td>50</td>
<td>51</td>
<td>52</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>60</td>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
<td>66</td>
<td>67</td>
<td>68</td>
<td>69</td>
<td>70</td>
<td>71</td>
<td>72</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td></td>
</tr>
<tr>
<td>76</td>
<td>77</td>
<td>78</td>
<td>79</td>
<td>80</td>
<td>81</td>
<td>82</td>
<td>83</td>
<td>84</td>
<td>85</td>
<td>86</td>
<td>87</td>
<td>88</td>
<td>89</td>
<td>90</td>
<td>91</td>
<td>92</td>
<td></td>
</tr>
<tr>
<td>93</td>
<td>94</td>
<td>95</td>
<td>96</td>
<td>97</td>
<td>98</td>
<td>99</td>
<td>100</td>
<td>101</td>
<td>102</td>
<td>103</td>
<td>104</td>
<td>105</td>
<td>106</td>
<td>107</td>
<td>108</td>
<td>109</td>
<td></td>
</tr>
<tr>
<td>110</td>
<td>111</td>
<td>112</td>
<td>113</td>
<td>114</td>
<td>115</td>
<td>116</td>
<td>117</td>
<td>118</td>
<td>119</td>
<td>120</td>
<td>121</td>
<td>122</td>
<td>123</td>
<td>124</td>
<td>125</td>
<td>126</td>
<td></td>
</tr>
<tr>
<td>127</td>
<td>128</td>
<td>129</td>
<td>130</td>
<td>131</td>
<td>132</td>
<td>133</td>
<td>134</td>
<td>135</td>
<td>136</td>
<td>137</td>
<td>138</td>
<td>139</td>
<td>140</td>
<td>141</td>
<td>142</td>
<td>143</td>
<td></td>
</tr>
</tbody>
</table>

### UNITS ON EDGE OF DISTRICT

<table>
<thead>
<tr>
<th>No</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td>28</td>
<td>29</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
<td>40</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
<td>50</td>
<td>51</td>
<td>52</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td>59</td>
<td>60</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
<td>66</td>
<td>67</td>
<td>68</td>
<td>69</td>
<td>70</td>
<td>71</td>
<td>72</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td>76</td>
<td>77</td>
<td></td>
</tr>
<tr>
<td>78</td>
<td>79</td>
<td>80</td>
<td>81</td>
<td>82</td>
<td>83</td>
<td>84</td>
<td>85</td>
<td>86</td>
<td>87</td>
<td>88</td>
<td>89</td>
<td>90</td>
<td>91</td>
<td>92</td>
<td>93</td>
<td>94</td>
<td></td>
</tr>
<tr>
<td>95</td>
<td>96</td>
<td>97</td>
<td>98</td>
<td>99</td>
<td>100</td>
<td>101</td>
<td>102</td>
<td>103</td>
<td>104</td>
<td>105</td>
<td>106</td>
<td>107</td>
<td>108</td>
<td>109</td>
<td>110</td>
<td>111</td>
<td></td>
</tr>
<tr>
<td>112</td>
<td>113</td>
<td>114</td>
<td>115</td>
<td>116</td>
<td>117</td>
<td>118</td>
<td>119</td>
<td>120</td>
<td>121</td>
<td>122</td>
<td>123</td>
<td>124</td>
<td>125</td>
<td>126</td>
<td>127</td>
<td>128</td>
<td></td>
</tr>
<tr>
<td>129</td>
<td>130</td>
<td>131</td>
<td>132</td>
<td>133</td>
<td>134</td>
<td>135</td>
<td>136</td>
<td>137</td>
<td>138</td>
<td>139</td>
<td>140</td>
<td>141</td>
<td>142</td>
<td>143</td>
<td>144</td>
<td>145</td>
<td></td>
</tr>
</tbody>
</table>

### POPULATION

<table>
<thead>
<tr>
<th>No</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td>28</td>
<td>29</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
<td>40</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
<td>50</td>
<td>51</td>
<td>52</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td>59</td>
<td>60</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
<td>66</td>
<td>67</td>
<td>68</td>
<td>69</td>
<td>70</td>
<td>71</td>
<td>72</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td>76</td>
<td>77</td>
<td></td>
</tr>
<tr>
<td>78</td>
<td>79</td>
<td>80</td>
<td>81</td>
<td>82</td>
<td>83</td>
<td>84</td>
<td>85</td>
<td>86</td>
<td>87</td>
<td>88</td>
<td>89</td>
<td>90</td>
<td>91</td>
<td>92</td>
<td>93</td>
<td>94</td>
<td></td>
</tr>
<tr>
<td>95</td>
<td>96</td>
<td>97</td>
<td>98</td>
<td>99</td>
<td>100</td>
<td>101</td>
<td>102</td>
<td>103</td>
<td>104</td>
<td>105</td>
<td>106</td>
<td>107</td>
<td>108</td>
<td>109</td>
<td>110</td>
<td>111</td>
<td></td>
</tr>
<tr>
<td>112</td>
<td>113</td>
<td>114</td>
<td>115</td>
<td>116</td>
<td>117</td>
<td>118</td>
<td>119</td>
<td>120</td>
<td>121</td>
<td>122</td>
<td>123</td>
<td>124</td>
<td>125</td>
<td>126</td>
<td>127</td>
<td>128</td>
<td></td>
</tr>
<tr>
<td>129</td>
<td>130</td>
<td>131</td>
<td>132</td>
<td>133</td>
<td>134</td>
<td>135</td>
<td>136</td>
<td>137</td>
<td>138</td>
<td>139</td>
<td>140</td>
<td>141</td>
<td>142</td>
<td>143</td>
<td>144</td>
<td>145</td>
<td></td>
</tr>
</tbody>
</table>

### TOTAL CONTRIBUTION TO CRITERION = 584.1642

(continued)
### Educational Information Network

**EDUCATIONAL INFORMATION NETWORK**

**EDUCOM**

000 0072

**DISTRICT 17**

**L.A. CONGRESSIONAL REAPPORTIONMENT - BELOW**

### UNITS

<table>
<thead>
<tr>
<th>1079</th>
<th>1090</th>
<th>1041</th>
<th>1049</th>
<th>1057</th>
<th>1119</th>
<th>1110</th>
<th>1111</th>
</tr>
</thead>
</table>

### UNITS ON FACE OF DISTRICT

<table>
<thead>
<tr>
<th>1079</th>
<th>1090</th>
<th>1041</th>
<th>1049</th>
<th>1057</th>
<th>1119</th>
<th>1110</th>
<th>1111</th>
</tr>
</thead>
</table>

### POPULATION VARIATION

<table>
<thead>
<tr>
<th>VARIATION</th>
<th>SQRD VARIATION</th>
<th>WEIGHTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>-37685.0000</td>
<td>7174.1964</td>
<td>7174.1984</td>
</tr>
</tbody>
</table>

### PROPORTION VARIATION

<table>
<thead>
<tr>
<th>VARIATION</th>
<th>SQRD VARIATION</th>
<th>WEIGHTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>4402.0000</td>
<td>4465.1066</td>
<td>44.9611</td>
</tr>
</tbody>
</table>

### NO OF UNITS

<table>
<thead>
<tr>
<th>WP</th>
<th>DEPR</th>
<th>COMPACTNESS</th>
<th>WEIGHTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>102.0000</td>
<td>20.0000</td>
<td>80.0000</td>
<td></td>
</tr>
</tbody>
</table>

### TOTAL CONTRIBUTION TO CRITERION

<table>
<thead>
<tr>
<th>CRITERION</th>
<th>7299.1595</th>
</tr>
</thead>
</table>

### CHARACTERISTICS OF THE AREA

**L.A. CONGRESSIONAL REAPPORTIONMENT - BELOW**

**INITIAL**

<table>
<thead>
<tr>
<th>VARI</th>
<th>WEIGHT</th>
<th>VAR X WEIGHT</th>
</tr>
</thead>
<tbody>
<tr>
<td>EQUALITY</td>
<td>26741.7967</td>
<td>4876.0000</td>
</tr>
<tr>
<td>POLITICAL</td>
<td>1073.1372</td>
<td>20.0000</td>
</tr>
<tr>
<td>COMPACTNESS</td>
<td>1219.4993</td>
<td>4876.0000</td>
</tr>
</tbody>
</table>

**CRITERION = 26741.7967**

*continued*
<table>
<thead>
<tr>
<th>LOOP</th>
<th>MOVES</th>
<th>TRADES</th>
<th>COMPACTNESS</th>
<th>EQUALITY</th>
<th>&quot;POLITICAL&quot; CRITERION</th>
<th>NTH</th>
<th>NTT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>18</td>
<td>0</td>
<td>1794.000000</td>
<td>945.07413</td>
<td>945.07476</td>
<td>7747.13581</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>13</td>
<td>0</td>
<td>1374.000000</td>
<td>2178.37342</td>
<td>917.28209</td>
<td>6611.65690</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>0</td>
<td>1940.000000</td>
<td>87.30662</td>
<td>907.10067</td>
<td>7774.41024</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>0</td>
<td>1134.000000</td>
<td>284.87382</td>
<td>905.14487</td>
<td>6514.92265</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>0</td>
<td>1243.000000</td>
<td>171.17147</td>
<td>920.90072</td>
<td>6175.26928</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>0</td>
<td>1745.000000</td>
<td>96.37615</td>
<td>919.84026</td>
<td>5966.27463</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>26</td>
<td>0</td>
<td>1715.000000</td>
<td>249.03921</td>
<td>916.90578</td>
<td>5853.92979</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>17</td>
<td>0</td>
<td>1711.000000</td>
<td>50.27051</td>
<td>909.21667</td>
<td>5812.44119</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>17</td>
<td>0</td>
<td>1147.000000</td>
<td>48.24427</td>
<td>910.32627</td>
<td>5735.61060</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>13</td>
<td>0</td>
<td>1166.000000</td>
<td>69.77456</td>
<td>920.59534</td>
<td>5664.63160</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>15</td>
<td>0</td>
<td>1156.000000</td>
<td>66.44477</td>
<td>920.51687</td>
<td>5616.65513</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>16</td>
<td>0</td>
<td>1143.000000</td>
<td>51.00335</td>
<td>921.16476</td>
<td>5647.00832</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>7</td>
<td>0</td>
<td>1194.000000</td>
<td>48.85914</td>
<td>922.75029</td>
<td>5724.35971</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>13</td>
<td>0</td>
<td>1123.900000</td>
<td>45.64380</td>
<td>922.35721</td>
<td>5464.01011</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>17</td>
<td>0</td>
<td>1120.000000</td>
<td>47.31779</td>
<td>922.39363</td>
<td>5486.05184</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>4</td>
<td>0</td>
<td>1121.000000</td>
<td>42.00554</td>
<td>919.84049</td>
<td>5446.44969</td>
<td>0</td>
</tr>
<tr>
<td>17</td>
<td>11</td>
<td>0</td>
<td>1117.000000</td>
<td>42.87031</td>
<td>920.96207</td>
<td>5416.37328</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>6</td>
<td>0</td>
<td>1117.000000</td>
<td>42.87031</td>
<td>920.96207</td>
<td>5416.37328</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>6</td>
<td>0</td>
<td>1117.000000</td>
<td>42.87031</td>
<td>920.96207</td>
<td>5416.37328</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>5</td>
<td>0</td>
<td>1094.000000</td>
<td>64.48909</td>
<td>921.08888</td>
<td>5332.47908</td>
<td>0</td>
</tr>
<tr>
<td>21</td>
<td>14</td>
<td>0</td>
<td>1094.000000</td>
<td>42.14664</td>
<td>917.03900</td>
<td>5196.22784</td>
<td>0</td>
</tr>
<tr>
<td>22</td>
<td>12</td>
<td>0</td>
<td>1094.000000</td>
<td>42.14664</td>
<td>917.03900</td>
<td>5196.22784</td>
<td>0</td>
</tr>
<tr>
<td>23</td>
<td>17</td>
<td>0</td>
<td>1011.000000</td>
<td>21.63312</td>
<td>927.63102</td>
<td>4993.66080</td>
<td>0</td>
</tr>
<tr>
<td>24</td>
<td>9</td>
<td>0</td>
<td>994.000000</td>
<td>24.32799</td>
<td>920.70191</td>
<td>4941.95732</td>
<td>0</td>
</tr>
<tr>
<td>25</td>
<td>19</td>
<td>0</td>
<td>987.000000</td>
<td>24.17103</td>
<td>911.84194</td>
<td>4976.01297</td>
<td>0</td>
</tr>
</tbody>
</table>

continued
### Final Units

| Units | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
| 177   | 654 | 650 | 660 | 661 | 663 | 664 | 665 | 666 | 667 | 668 | 669 | 670 | 671 | 672 | 673 | 674 | 675 | 676 | 677 | 678 | 679 | 680 | 681 | 682 | 683 | 684 |

### Units on Foot of District

| Units | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
| 177   | 654 | 650 | 660 | 661 | 663 | 664 | 665 | 666 | 667 | 668 | 669 | 670 | 671 | 672 | 673 | 674 | 675 | 676 | 677 | 678 | 679 | 680 | 681 | 682 | 683 | 684 |

### Population Variation

<table>
<thead>
<tr>
<th>Population</th>
<th>Variation</th>
<th>Weighted</th>
</tr>
</thead>
<tbody>
<tr>
<td>44156,0000</td>
<td>499,0000</td>
<td>1,3637</td>
</tr>
</tbody>
</table>

### Proportion Variation

<table>
<thead>
<tr>
<th>Proportion</th>
<th>Weighted</th>
</tr>
</thead>
<tbody>
<tr>
<td>157,4152</td>
<td>45,4413</td>
</tr>
</tbody>
</table>

### Weighted Compactness

<table>
<thead>
<tr>
<th>Weighted Compactness</th>
<th>Total Contribution to Criterion</th>
</tr>
</thead>
<tbody>
<tr>
<td>259,3150</td>
<td></td>
</tr>
</tbody>
</table>

---

**continued**
000 0072

1065
1052
1074
laRq
1107

ifirq

-1081

ar.011

wP

-gels

954
1015

-A113-

1101
1124
1144

1143

1142

1102
1125
1145

1177-107A-

1100
1121

1099

gac
1037

1475-1076

1136

13
11

12
39

it
31

34

14

146
668
1279

4

145
667
1278

3

907
1050
10791103
1126
1146

-998
1051
1084
1114
1127
1147

46

16

1242

1281
15
37

164

6

149
670

-96---

5

1148

1149

999 "1.66Y
1052 1053
1085 1086
1105 1106

1283
17
47

673
1284
18
48

8

672-

=

.

1364
1114

1092
1113
1119

_..1078

1150

TO-0.

1054
1047
1107

-

101
169
674
1285
19
49

-

35

1046
1070_
1060

.

.4637

38
125
172
1268
1288
22
52
.

1114
1134
1082

Iffin

1026
1058

1289
23
53

39
126
173
1269

42.0000

COMPACTNESS

4232.1798

.

25
55

41
134
175
1271
1291

_

1054
1114

1061
1085
1096
1116

1.0?

1047

1027 1028
1059 1060
101)2--1143
1115 1116
1135 1136
1083 110q

40
133
174
1270
1290
24
54

SQRD VARIATION

-1025
1155 1956 1057
1088 1089 1090
1108 1112 1113
fili---11M--It33
1072 1080 1081

21
_ 51

36
119
171
f76
12 7

1053
1106

1115
1111

SORD VARIATION

1037
1105

1114
1110

-1.-91-1r--1024

20
50

117
170
675
1286

710.7855

100.0001

DESPR

165.0552

PROPORTION

-.6810

-

10Ta_los.s.

1044
1045
1068 --_1069
---1-658
1.09

1029
1061
1044
1117
1137
1110

42
139
176
1272
1292
26
56

43

1118
1138
1111

1195

1030
1063

140
178
1273
1293
27
57

168.0000

WEfGHttb

42.3218

WEIGHTED

-.4637

WEIGHTED

_1115

1058

1048
1037
1063
1086
1104
1117

L.A. CONGPsSIONAL REAPPORTIONMENT - BELOW

PCT VARIATION

1044
1096

100
166

7

_

1112
1081

ficli

1077_

-1054

1041
1067

99
165

TOTAL L`NTRIBUTYON TO -ffITERION

6*.(1000

95
148
669
1280

1.-122

-298.0000

42115.0090

'V+

1034

la41
1094

liNKIATION

_

69513.1000

1039
1089

1040
1066
1051
1975
1090
1118
1082

a

111

1138
1175

_

A

-

93

1

-

1118
1074
1117

92
14
666
1277

-

1106
198Q

111e8

1164
1051
1173

-10F'k

17

UNITS
_ OM Ennr
_ - OF DISTRICT

1072-

1162
1087
1105

1650-

IOIR
1049

UNITS

0ISIRICT
FINAL
_

1031
1062
1096
1119
1139

1120
1140

I-097

1032
1073

45
142
662
1275
1297
2829
58
59

44
141
179
1274
1294

1033
1074
1098
1121
1141

10
30
60

75
143
665
1276

000 0072


<table>
<thead>
<tr>
<th>CHARACTERISTIC OF THE AREA</th>
<th>L.T. CONGRESSIONAL REAPPORTIONMENT - BELOW</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FINAL</td>
</tr>
<tr>
<td></td>
<td>TOTAL VAP</td>
</tr>
<tr>
<td></td>
<td>WEIGHT</td>
</tr>
<tr>
<td></td>
<td>VAR X WEIGHT</td>
</tr>
<tr>
<td>EQUALITY</td>
<td>32.17103</td>
</tr>
<tr>
<td></td>
<td>1.00000</td>
</tr>
<tr>
<td></td>
<td>32.17103</td>
</tr>
<tr>
<td>POLITICAL</td>
<td></td>
</tr>
<tr>
<td>COMPACTNESS</td>
<td>933.00700</td>
</tr>
<tr>
<td></td>
<td>6.00000</td>
</tr>
<tr>
<td></td>
<td>7032.00000</td>
</tr>
</tbody>
</table>

**CUTTERION:** 4476.91247

**DATA ERRORS:**

- 1247 17 1 64 1 2
COST ESTIMATE

Computer charges are calculated at the rate of $150./computational unit (hour). The sample run consumed 7 minutes of computer time, which should be typical for large jobs of similar complexity.

Therefore, for the sample run shown above,

Approximate charge to user = computer time + postage, handling, and consulting + network overhead

= $17.50 + $7.50 + network overhead

= $25.00 + network overhead

CONTENTS—T8 FSU BELOW

pages  Identification & Abstract
1-4        User Instructions
5-10       I/O
11-18      Cost—Contents
DESCRIPITIVE TITLE  Equipercentile Equating Program

CALLING NAME  SCORMACH

INSTALLATION NAME  Wharton Computational Services
University of Pennsylvania

AUTHOR(S) AND AFFILIATION(S)  Daniel Ashler, Director
Wharton Computational Services
University of Pennsylvania

LANGUAGE  FORTRAN IV

COMPUTER  IBM System 360/75

PROGRAM AVAILABILITY  Proprietary; usage permitted but program deck or listing not available

CONTACT  Daniel Ashler, Wharton Computational Services, University of Pennsylvania, Philadelphia, Pa. 19104
Tel: (215) 594-6422

FUNCTIONAL ABSTRACT

SCORMACH uses the equipercentile equating method to obtain "comparable" scores (having identical means, standard deviations, and distributions) for several forms of a given test or for different tests. A given score on one test is considered comparable to a given score on another if, in the two (not necessarily distinct) groups of examinees, the same proportions attain less than the respective given scores—i.e., if a score on one test has the same percentile rank as its comparable score on the other test.

The principal output is a table for each pair of tests by which, given a regressed equipercentile score on either test, one can determine the corresponding regressed equipercentile score on the other. If desired, the same information may also be had in the form of a graph produced on a CalComp plotter. Other tables and graphs allow comparisons of raw scores with raw scores, raw scores with regressed scores, etc.

The scores that are of interest here are the examinees' "true scores" on each respective test. Although true scores cannot be observed directly, the characteristics of the distribution continue
of true scores for a given test can be estimated. This estimation, resulting in "regressed" scores, is made by modifying the value of each observed score in accordance with the reliability coefficient of the test.\(^1\) One technique for equating scores is to plot points that correspond to pairs of comparable regressed scores on rectangular coordinates. A smooth curve is drawn through the points; pairs of equated scores are read from this curve. In addition to equating scores, the program predicts scores on one test from raw scores on the other. The predicted score is the regressed score that has the same percentile rank as the predictor raw score on the other test.

The program accepts a set of raw scores and a coefficient of reliability for each of two or more tests. (There is no limit to the number of tests to be equated in a single run.) The scores of the first test (anchor test) entered as input are equated with the scores of each subsequent test, one test at a time, by the equipercentile method described above.

The comparability of the scores of different forms (or tests) is specific to the type of groups used in obtaining them. A group that is both representative (of the population of examinees for whom the tests are intended) and large (at least 500 examinees) should be used. Preferably, the same examinees should take all tests to be equated (Ref. 1, p. 758). Although the value of the results is enhanced when all tests are taken by the same examinees and by an equal number of examinees, the program does not require that these conditions be met.

REFERENCES

USER INSTRUCTIONS

Preparation of Input

System Cards

Wharton Computational Services will prepare the necessary system cards to conform with the current version of the operating system. A cover letter should accompany the data stating whether CalComp output is desired.

Data-Input Cards

A. Provide a score packet for each test to be analyzed, consisting of a Title Card followed by Score Cards for that test. Data for the "anchor" test should come first. Scores of each subsequent test are equated with those of the "anchor" test. There is no limit to the number of tests for which score packets are submitted.

(1) Title Card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-8</td>
<td>Alphabetic title of test. It may <em>not</em> include any numeric digits, periods, plus signs or minus signs</td>
</tr>
<tr>
<td>9-80</td>
<td>Ignored by the program</td>
</tr>
</tbody>
</table>

(2) Score Cards

Punch as many test scores as desired on each card. Scores must be separated by at least one blank. When Col. 80 of a card is reached, continue in Col. 1 of the next card. (If a score ends in Col. 80, begin the next score in Col. 2 of the next card to have a blank between scores.) The scores need not be in ascending order *but* computer time will be saved if they are. If a score is repeated N times, the score need not be punched N times. Instead, punch the value of N followed by an asterisk followed by the score. For example, if 17 people have the score 112.5, punch 17*112.5. Follow the last score with a blank, a semicolon, at least one more blank, and then the reliability coefficient of the test in the form

```
XXXXXXXX RELIABILITY IS 0.RRR
```

where XXXXXXX represents the test title (exactly as punched in the Title Card) and 0.RRR is the reliability coefficient of the test. *(Note: The letter Ø is slashed; the number zero is not slashed.)*

*continued*
B. SCORMACH can also process (in the same run) additional sets of score packets, each of which consists of an "anchor" packet followed by as many other packets as desired. Such additional sets of score packets should be wrapped separately and clearly marked; the anchor packet must be the first packet in each set.

Description of Output

Output from SCORMACH appears in both tables and graphs. For each test, the tables include

(a) an "echo print" of the raw scores of the examinees. This may be omitted if the user so specifies in the cover letter. (It is not included in Sample Output.)

(b) a listing of every raw score attained by the examinees in the group, each with its corresponding percentile rank and regressed score

(c) a listing of the regressed score on the other test that corresponds to each raw score attained by the examinees in the group

(d) a listing of the corresponding raw and regressed scores for each integer percentile rank

The graphs include

(a) the percentile ranks of the raw and regressed scores of both tests

(b) the predicted scores of each test, given the raw scores of the other test

(c) the raw scores of both tests at integer percentile values

(d) the regressed equipercentile scores—including, regressed scores of test one versus comparable regressed scores of test two

If CalComp plotter output is specified, a graph is produced like the final illustration.
SAT VERB RELIABILITY IS 0.91

SAT MATH RELIABILITY IS 0.89
### Sample Output

<table>
<thead>
<tr>
<th>SAT VEBR</th>
<th>RAW SCORE</th>
<th>ZILE RANK</th>
<th>REGRESSED SCORE</th>
<th>RAW SCORE</th>
<th>ZILE RANK</th>
<th>REGRESSED SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>22.00</td>
<td>38.12</td>
<td>10.51</td>
<td>93.198</td>
<td>96.15</td>
<td>47.740</td>
<td>92.04</td>
</tr>
<tr>
<td>23.00</td>
<td>39.00</td>
<td>38.41</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>24.00</td>
<td>40.00</td>
<td>39.33</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>25.00</td>
<td>41.00</td>
<td>40.26</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>26.00</td>
<td>42.00</td>
<td>41.19</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>27.00</td>
<td>43.00</td>
<td>42.14</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>28.00</td>
<td>44.00</td>
<td>43.09</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>29.00</td>
<td>45.00</td>
<td>44.05</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>30.00</td>
<td>46.00</td>
<td>45.00</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>31.00</td>
<td>47.00</td>
<td>46.91</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>32.00</td>
<td>48.00</td>
<td>48.86</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>33.00</td>
<td>49.00</td>
<td>50.81</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>34.00</td>
<td>50.00</td>
<td>52.87</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
<tr>
<td>35.00</td>
<td>51.00</td>
<td>54.93</td>
<td>94.207</td>
<td>97.17</td>
<td>48.740</td>
<td>92.04</td>
</tr>
</tbody>
</table>

### SAT MATH

<table>
<thead>
<tr>
<th>RAW SCORE</th>
<th>ZILE RANK</th>
<th>REGRESSED SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>23.00</td>
<td>36.00</td>
<td>40.25</td>
</tr>
<tr>
<td>24.00</td>
<td>37.00</td>
<td>41.20</td>
</tr>
<tr>
<td>25.00</td>
<td>38.00</td>
<td>42.15</td>
</tr>
<tr>
<td>26.00</td>
<td>39.00</td>
<td>43.10</td>
</tr>
<tr>
<td>27.00</td>
<td>40.00</td>
<td>44.05</td>
</tr>
<tr>
<td>28.00</td>
<td>41.00</td>
<td>45.00</td>
</tr>
<tr>
<td>29.00</td>
<td>42.00</td>
<td>46.00</td>
</tr>
<tr>
<td>30.00</td>
<td>43.00</td>
<td>47.00</td>
</tr>
<tr>
<td>31.00</td>
<td>44.00</td>
<td>48.00</td>
</tr>
<tr>
<td>32.00</td>
<td>45.00</td>
<td>49.00</td>
</tr>
<tr>
<td>33.00</td>
<td>46.00</td>
<td>50.00</td>
</tr>
<tr>
<td>34.00</td>
<td>47.00</td>
<td>51.00</td>
</tr>
<tr>
<td>35.00</td>
<td>48.00</td>
<td>52.00</td>
</tr>
<tr>
<td>36.00</td>
<td>49.00</td>
<td>53.00</td>
</tr>
<tr>
<td>37.00</td>
<td>50.00</td>
<td>54.00</td>
</tr>
<tr>
<td>38.00</td>
<td>51.00</td>
<td>55.00</td>
</tr>
<tr>
<td>39.00</td>
<td>52.00</td>
<td>56.00</td>
</tr>
<tr>
<td>40.00</td>
<td>53.00</td>
<td>57.00</td>
</tr>
<tr>
<td>41.00</td>
<td>54.00</td>
<td>58.00</td>
</tr>
<tr>
<td>42.00</td>
<td>55.00</td>
<td>59.00</td>
</tr>
</tbody>
</table>

### Notes
- **SAT VEBR** and **SAT MATH** columns represent the raw scores and the regressed scores.
- **ZILE RANK** column shows the rank based on Z-scores.
- **RAW SCORE** and **REGRESSED SCORE** are the raw and regressed scores respectively.

**Continued...**
RESULT OF INTERPOLATION AND EXTRAPOLATION FOR 50 SAT VERB VALUES AND 49 SAT MATH VALUES

<table>
<thead>
<tr>
<th>RAW SAT VERB</th>
<th>REGRESSED SAT VERB</th>
<th>%ILE</th>
<th>PREDICTED SAT VERB</th>
<th>RAW SAT MATH</th>
<th>REGRESSED SAT MATH</th>
<th>%ILE</th>
<th>PREDICTED SAT MATH</th>
</tr>
</thead>
<tbody>
<tr>
<td>22.0000</td>
<td>23.0594</td>
<td>0.0657</td>
<td>24.3585</td>
<td>23.0000</td>
<td>24.3585</td>
<td>0.0657</td>
<td>23.3594</td>
</tr>
<tr>
<td>23.0000</td>
<td>24.0134</td>
<td>0.3285</td>
<td>27.3773</td>
<td>27.0000</td>
<td>28.1320</td>
<td>0.3942</td>
<td>26.6903</td>
</tr>
<tr>
<td>25.0000</td>
<td>25.9212</td>
<td>0.5913</td>
<td>28.4151</td>
<td>28.0000</td>
<td>29.0754</td>
<td>1.0512</td>
<td>27.4475</td>
</tr>
<tr>
<td>26.0000</td>
<td>26.8752</td>
<td>0.8541</td>
<td>28.7924</td>
<td>29.0000</td>
<td>30.0188</td>
<td>1.5769</td>
<td>29.2777</td>
</tr>
<tr>
<td>27.0000</td>
<td>27.8291</td>
<td>1.1827</td>
<td>29.3113</td>
<td>30.0000</td>
<td>30.9622</td>
<td>2.9711</td>
<td>30.9635</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td></td>
<td>.</td>
<td>.</td>
<td>.</td>
<td></td>
<td>.</td>
</tr>
<tr>
<td>68.0000</td>
<td>66.9406</td>
<td>98.0830</td>
<td>67.9434</td>
<td>72.0000</td>
<td>70.5849</td>
<td>99.4087</td>
<td>68.2761</td>
</tr>
<tr>
<td>69.0000</td>
<td>67.8945</td>
<td>99.2773</td>
<td>69.6415</td>
<td>73.0000</td>
<td>71.5283</td>
<td>99.6058</td>
<td>68.8484</td>
</tr>
<tr>
<td>70.0000</td>
<td>68.8485</td>
<td>99.6058</td>
<td>71.5283</td>
<td>75.0000</td>
<td>73.4151</td>
<td>99.8029</td>
<td>69.1024</td>
</tr>
<tr>
<td>71.0000</td>
<td>69.0024</td>
<td>99.8029</td>
<td>73.4151</td>
<td>76.0000</td>
<td>74.3585</td>
<td>99.9343</td>
<td>71.1713</td>
</tr>
<tr>
<td>73.0000</td>
<td>71.7103</td>
<td>99.9343</td>
<td>74.3585</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>

VALUES COMPUTED FOR INTEGER PERCENTILE RANKS

<table>
<thead>
<tr>
<th>%ILE</th>
<th>RAW SAT VERB</th>
<th>REGRESSED SAT VERB</th>
<th>RAW SAT MATH</th>
<th>REGRESSED SAT MATH</th>
<th>%ILE</th>
<th>RAW SAT VERB</th>
<th>REGRESSED SAT VERB</th>
<th>RAW SAT MATH</th>
<th>REGRESSED SAT MATH</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>26.444</td>
<td>27.299</td>
<td>27.922</td>
<td>29.002</td>
<td>51</td>
<td>45.069</td>
<td>45.066</td>
<td>47.521</td>
<td>47.492</td>
</tr>
<tr>
<td>2</td>
<td>30.349</td>
<td>31.023</td>
<td>30.063</td>
<td>31.022</td>
<td>52</td>
<td>45.319</td>
<td>45.304</td>
<td>47.709</td>
<td>47.669</td>
</tr>
<tr>
<td>3</td>
<td>31.666</td>
<td>32.280</td>
<td>31.509</td>
<td>32.386</td>
<td>53</td>
<td>45.566</td>
<td>45.542</td>
<td>47.897</td>
<td>47.846</td>
</tr>
<tr>
<td>4</td>
<td>32.617</td>
<td>33.188</td>
<td>32.410</td>
<td>33.255</td>
<td>54</td>
<td>45.818</td>
<td>45.780</td>
<td>48.089</td>
<td>48.028</td>
</tr>
<tr>
<td>5</td>
<td>33.479</td>
<td>34.010</td>
<td>33.416</td>
<td>34.203</td>
<td>55</td>
<td>46.055</td>
<td>46.007</td>
<td>48.287</td>
<td>48.214</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>46</td>
<td>44.042</td>
<td>44.086</td>
<td>46.628</td>
<td>46.649</td>
<td>96</td>
<td>62.008</td>
<td>61.225</td>
<td>63.595</td>
<td>62.696</td>
</tr>
<tr>
<td>47</td>
<td>44.245</td>
<td>44.279</td>
<td>46.795</td>
<td>46.807</td>
<td>97</td>
<td>63.043</td>
<td>62.211</td>
<td>64.695</td>
<td>63.693</td>
</tr>
<tr>
<td>48</td>
<td>44.447</td>
<td>44.473</td>
<td>46.962</td>
<td>46.964</td>
<td>98</td>
<td>65.312</td>
<td>64.376</td>
<td>66.366</td>
<td>65.270</td>
</tr>
<tr>
<td>49</td>
<td>44.650</td>
<td>44.666</td>
<td>47.145</td>
<td>47.137</td>
<td>99</td>
<td>68.297</td>
<td>67.224</td>
<td>69.556</td>
<td>68.279</td>
</tr>
<tr>
<td>50</td>
<td>44.853</td>
<td>44.860</td>
<td>47.333</td>
<td>47.314</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>
continued
COST ESTIMATE

For each separate computer run of SCORMACH, there will be a setup charge of $50.00 plus a per-student charge of 10¢ for each test beyond the "anchor" test. Postage, handling, and consulting fees are included. And additional $20.00 charge will be made for each CalComp plot. (If CalComp option is requested, one plot will be produced for each pair of tests equated.)

Charge to user = $50.00 + 10¢/student for each test beyond "anchor" + network overhead

CONTENTS—SCORMACH

pages
1–2 Identification & Abstract
3–4 User Instructions
5–9 I/O
11 Cost—Contents
Multiple Scalogram Analysis

DESCRIPTIVE TITLE

CALLING NAME  MSA

INSTALLATION NAME  Office of Computational Sciences
Educational Testing Service

AUTHOR(S) AND
AFFILIATION(S)

Procedure due to: L. Guttman
J. Lingoes
University of Michigan

Program due to: J. Lingoes
University of Michigan

Adaptation due to: D. Kirk
Educational Testing Service

Advisor on use: H. Harman
Educational Testing Service

LANGUAGE  MAP

COMPUTER  IBM 360/65

PROGRAM AVAILABILITY  Decks and listings presently available

CONTACT  Mr. Ernest Anastasio, Office of Data
Analysis Research, Educational Testing Service,
Rosedale Road, Princeton, N.J. 08540
Tel.: (609) 921-9000 ext. 2552

FUNCTIONAL ABSTRACT

This program performs a multiple scalogram analysis using the
method of James Lingoes¹. Do not confuse this with Guttman-
Lingoes Multidimensional scalogram analysis which is available
under the name of MSA-I (EIN Abstract 000 0070).

REFERENCES

1. Lingoes, J.C., "A Set-Theoretic Model for Analyzing Dichoto-

10/70
USER INSTRUCTIONS

Input Deck

Parameter Card A

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-3</td>
<td>Number of cases to be run</td>
</tr>
</tbody>
</table>

Title Card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2-72</td>
<td>Any title</td>
</tr>
</tbody>
</table>

Parameter Card B

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-3</td>
<td>Number of subjects (&lt;100)</td>
</tr>
<tr>
<td>4-6</td>
<td>Number of items (&lt;122)</td>
</tr>
<tr>
<td>7</td>
<td>1: responses must be dichotomized</td>
</tr>
<tr>
<td></td>
<td>0: responses are either 1 or 0</td>
</tr>
<tr>
<td>8-10</td>
<td>Reproduceability criterion (normally about .80)</td>
</tr>
<tr>
<td>11-16</td>
<td>Chi-square criterion (normally 10.827)</td>
</tr>
<tr>
<td>17-20</td>
<td>ID run number (may be alphameric)</td>
</tr>
<tr>
<td>21</td>
<td>1: print input data</td>
</tr>
<tr>
<td></td>
<td>0: do not print input data</td>
</tr>
<tr>
<td>22</td>
<td>1: print debugging output</td>
</tr>
<tr>
<td></td>
<td>0: do not print debugging output</td>
</tr>
<tr>
<td>23</td>
<td>1: Description Cards present</td>
</tr>
<tr>
<td></td>
<td>0: no Description Cards</td>
</tr>
<tr>
<td>24</td>
<td>1: Cornell scores printed for scales with two or more items</td>
</tr>
<tr>
<td></td>
<td>0: Cornell scores not printed</td>
</tr>
</tbody>
</table>

Parameter Card C (Omit if Col. 7 of Parameter Card B is 0.)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-3</td>
<td>Maximum number of categories for any item</td>
</tr>
</tbody>
</table>

continued
Parameter Card D (Omit if number punched on Parameter Card C > 2.)

<table>
<thead>
<tr>
<th>Column</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–3</td>
<td>Response code for item 1, response A</td>
</tr>
<tr>
<td>4–6</td>
<td>Response code for item 1, response B</td>
</tr>
<tr>
<td>7–9</td>
<td>Response code for item 2, response A</td>
</tr>
<tr>
<td>·</td>
<td></td>
</tr>
<tr>
<td>76–79</td>
<td>Response code for item 13, response B</td>
</tr>
</tbody>
</table>

Note: This card is used when number on Parameter Card C is 2. Punch as many cards as required to satisfy the number of items specified in Parameter Card B (Col. 4–6).

Format Card
Standard FORTRAN format for input data. Omit the word FORMAT.

Description Deck (Omit this deck if Col. 23 of Parameter Card B is 0.)

<table>
<thead>
<tr>
<th>Column</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–72</td>
<td>Alphameric description card. Punch one for each item</td>
</tr>
</tbody>
</table>

Data Deck
Punch one subject ID followed by all item codes for the number of items specified according to the format specified on the Format Card.

Multiple case runs will repeat the Input Deck from Title Card through Data Deck, as many times as specified on Parameter Card A.

Job Deck
The program is currently in the system and may be called as follows.

//JOB CARD (will be provided by ETS personnel).
//EXEC GITNGO,NAME=MSA
//GO,SYSIN DD #
(Insert Input Deck or Decks here.)

continued
Output

The output will vary according to the options selected. The Sample Input/Output gives a hypothetical example with 8 items and 25 subjects. The output selections included input parameters, Cornell scores and answer patterns ordered by subject ID for scales 1 and 2, and subjects ordered by Cornell scores for scales 1 and 2.
SAMPLE INPUT

```plaintext
//MSATST  JOB (....
//EXEC GITNGO,NAME=MSA
//GO.SYSIN  DD *
1
1 THIS WILL TEST MSA WITH A HYPOTHETICAL 2-DIMENSIONAL CASE
0250080.80 3.84TEST1011
(12,8(2X,11))
  ITEM NO. 1
  ITEM NO. 2
  ITEM NO. 3
  ITEM NO. 4
  ITEM NO. 5
  ITEM NO. 6
  ITEM NO. 7
  ITEM NO. 8
01   1 0 0 1 0 0 1 1
02   0 0 1 1 0 1 0 0
03   1 0 0 0 1 1 0 0
04   1 0 1 1 0 1 1 0
05   1 1 0 0 1 1 0 1
06   0 0 1 1 0 1 1 0
07   1 0 0 0 0 1 0 0
08   1 0 0 0 1 0 0 1
09   1 0 1 0 1 1 0 0
10   1 0 0 1 1 1 0 1
11   1 0 0 1 1 0 0 1
12   1 0 0 1 1 1 0 0
13   0 1 1 0 1 1 0 0
14   1 0 0 1 0 1 0 1
15   1 1 1 0 1 1 0 1
16   1 0 0 1 0 1 1 0
17   1 1 0 0 1 0 0 1
18   0 0 1 1 1 1 0 0
19   1 0 0 0 1 1 0 1
20   1 0 1 1 0 1 0 0
21   1 0 0 1 0 0 0 1
22   1 0 0 1 0 1 1 1
23   1 1 0 0 1 1 0 0
24   1 0 1 1 1 1 0 0
25   0 0 1 0 1 1 0 0
/ *
```

ITEM NO. 1
ITEM NO. 2
ITEM NO. 3
ITEM NO. 4
ITEM NO. 5
ITEM NO. 6
ITEM NO. 7
ITEM NO. 8
01   1 0 0 1 0 0 1 1
02   0 0 1 1 0 1 0 0
03   1 0 0 0 1 1 0 0
04   1 0 1 1 0 1 1 0
05   1 1 0 0 1 1 0 1
06   0 0 1 1 0 1 1 0
07   1 0 0 0 0 1 0 0
08   1 0 0 0 1 0 0 1
09   1 0 1 0 1 1 0 0
10   1 0 0 1 1 1 0 1
11   1 0 0 1 1 0 0 1
12   1 0 0 1 1 1 0 0
13   0 1 1 0 1 1 0 0
14   1 0 0 1 0 1 0 1
15   1 1 1 0 1 1 0 0
16   1 0 0 1 0 1 1 0
17   1 1 0 0 1 0 0 1
18   0 0 1 1 1 1 0 0
19   1 0 0 0 1 1 0 1
20   1 0 1 1 0 1 0 0
21   1 0 0 1 0 0 0 1
22   1 0 0 1 0 1 1 1
23   1 1 0 0 1 1 0 0
24   1 0 1 1 1 1 0 0
25   0 0 1 0 1 1 0 0
/ *
SAMPLE OUTPUT

THIS WILL TEST MSA WITH A HYPOTHETICAL 2-DIMENSIONAL CASE

ORIGINAL DATA

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

SCALE NO. INCLUDES THE FOLLOWING ITEMS

<table>
<thead>
<tr>
<th>ITEM NO.</th>
<th>REFLECTED</th>
<th>MARGINAL SUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>15</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>5</td>
</tr>
</tbody>
</table>

CHI-SQUARE CRITERION = 3.840

PHI = 0.80

REPRODUCIBILITY = 1.000

MIN. MARGINAL REPR. = 0.700

NUMBER IN SCALE = 4

continued
### Cornwell Scores and Answer Patterns Ordered

BY SUBJECT IDENTIFICATION NUMBER

FOR SCALE NO. 1

<table>
<thead>
<tr>
<th>ID NUMBER</th>
<th>CORNELL SCORE</th>
<th>ANSWER PATTERN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>11</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>14</td>
<td>3</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>16</td>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>17</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>18</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>19</td>
<td>3</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>21</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>22</td>
<td>3</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>23</td>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>25</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
</tbody>
</table>

COLUMN SUMS FOR THIS SCALE

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>15</td>
<td>10</td>
<td>5</td>
</tr>
</tbody>
</table>

### Subjects Ordered by Cornwell Scores

FOR SCALE NO. 1

<table>
<thead>
<tr>
<th>CORNELL SCORE</th>
<th>ID NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
</tr>
<tr>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>19</td>
</tr>
<tr>
<td>4</td>
<td>21</td>
</tr>
<tr>
<td>4</td>
<td>17</td>
</tr>
<tr>
<td>4</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
</tr>
</tbody>
</table>
**SCALE NO. 2**

**includes the following items**

<table>
<thead>
<tr>
<th>Item No.</th>
<th>Reflected</th>
<th>Marginal Sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>5</td>
</tr>
</tbody>
</table>

**Chi-Square Criterion** = 3.840

**Phi** = 0.80

**Reproducibility** = 1.000

**Min. Marginal Repr.** = 0.700

**Number in Scale** = 4

---

**Cornell Scores and Answer Patterns Ordered by Subject Identification Number**

**For Scale No. 2**

<table>
<thead>
<tr>
<th>ID Number</th>
<th>Cornell Score</th>
<th>Answer Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>2 4 5 7</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>1 1 1 0</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>15</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>17</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>18</td>
<td>4</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>19</td>
<td>5</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>21</td>
<td>1</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>22</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>23</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>25</td>
<td>0</td>
<td>0 0 0 0</td>
</tr>
</tbody>
</table>

**Column Sums for This Scale**

20 15 10 5

---

*continued*
### Subjects Ordered by Cornell Scores

For Scale No. 2

<table>
<thead>
<tr>
<th>Cornell Score</th>
<th>ID Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>0</td>
<td>22</td>
</tr>
<tr>
<td>0</td>
<td>23</td>
</tr>
<tr>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>1</td>
<td>14</td>
</tr>
<tr>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>19</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>17</td>
</tr>
<tr>
<td>4</td>
<td>18</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
</tr>
</tbody>
</table>

All possible scales have been formed—Multiple Scalogram

Analysis is finished for job no. test
COST ESTIMATE

The pricing algorithm for the Educational Testing Service is based on a complex formula involving the amount of core usage, card-processing, tape and dedicated device usage, lines printed and central processor unit (CPU) elapsed time. Computer costs for the job included in the Sample Input were $1.41.

Charge to user = computer costs + postage and handling + network overhead

= $1.41 + $5.00 + network overhead

= $6.41 + network overhead

CONTENTS—MSA

pages
1 Identification & Abstract
3–5 User Instructions
7–11 I/O
13 Cost—Contents
Nonmetric Multidimensional Scaling

Subroutine KRUSCAL

Michigan State University
Computer Laboratory

Alan M. Lesgold
Computer Institute for Social Science Research
Michigan State University

FORTRAN or COMPASS

CDC 3600

Decks and listings presently available

Dr. Anders Johanson, Programming Supervisor
Applications Programming, Computer Laboratory, Computer Center, Michigan State University, East Lansing, Mich. 48823
Tel.: (517) 355-4684

KRUSCAL is an implementation of J.B. Kruskal's recently published\(^1\) numerical method for multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis\(^2\). Given a matrix of similarities or dissimilarities between \(n\) variables, the routine outputs a configuration of \(n\) points in a specified number of dimensions such that the distance between any two points is a monotone function of the dissimilarity of the two variables corresponding to those two points.

General Description of Input

KRUSCAL will accept as input any matrix or halfmatrix (below major diagonal) with or without the major diagonal, of similarities or dissimilarities, including, among others, correlation coefficients, confusion probabilities, interaction rates among groups, etc. This matrix need not be symmetric, and the program allows for missing data; interpoint distances corresponding to missing data values do not contribute to the stress.

continued
General Description of Output

The output consists of three sections for the two-dimensional case and two sections for other dimensionalities. In every case, a history of computation is printed, showing for each iteration the following information.

\[ \text{STRESS} = \text{Normalized stress achieved} \]
\[ \text{SRAT} = \text{The rate of stress improvement} = \frac{\text{STRESS}_i}{\text{STRESS}_{i-1}} \]
\[ \text{SRATAV} = \text{Weighted average of SRAT} \]
\[ = \text{SRAT}_i^{\frac{1}{3}} \times \text{SRATAV}_{i-1}^{\frac{2}{3}} \]
\[ \text{CAGRGL} = \text{Cosine of angle between gradient and previous gradient} \]
\[ \text{COSAV} = \text{Weighted average of CAGRGL} \]
\[ = \left(\frac{1}{3}\right)\text{CAGRGL}_i + \left(\frac{2}{3}\right)\text{COSAV}_{i-1} \]
\[ \text{ACSAV} = \text{Weighted average of the magnitude of CAGRGL, computed in the same manner as COSAV} \]
\[ \text{SFGR} = \text{Scale factor of gradient, [same as Kruskal's mag(g)]} \]
\[ \text{STEP} = \text{Step size} \]

Also, in every case, the final number of dimensions is printed. There is an option for printing the interpoint distances in each final configuration, and an option for punching the final configuration onto cards (see description of option cards).

In the two-dimensional case, the final configuration is also plotted (on the line printer, not on the plotter).

REFERENCES


USER INSTRUCTIONS

Calling Sequence
The routine is called in the ordinary manner for calling FORTRAN SUBROUTINES that have no formal parameters.1, 2

From a FORTRAN program:
CALL KRUSCAL

From a COMPASS program:
BRTJ ($)KRUSCAL, *
SLJ *+1
00 DICT.

Arguments or Parameters
All parameters are read by SUBROUTINE KRUSCAL from the data deck. See Description of Input.

Space Required (Decimal)
The subroutine itself and the subroutines that it contains take a total of 13817 words of core storage. The total amount of core storage taken up by KRUSCAL and all of the routines (including library routines) that it calls is 17285 words.

Error Returns or Error Codes
The message, KRUSCAL, IMPROPER CONTROL CARD, is printed along with the contents of any control card in the data deck that is incorrect. The program then terminates by doing a return to SCOPE through Q8QERROR.

Error Stop
A condition that is logically impossible in the program (i.e., a recognized machine error) will generate the message KRUSCAL and will terminate through Q8QERROR.

Accuracy
Accuracy is not a problem, since the original data values are stored, and the measures used in determining when a solution has been reached are functions, in part, of these original data values.

Caution to Users: Up to 60 variables may be scaled in up to 10 dimensions. Users should be aware of the "local minima" problem.3

continued
Flow Chart

The algorithm and logical flow of the program are described in Ref. 3. The following array names have been changed from those recorded in that reference.

<table>
<thead>
<tr>
<th>In Ref. 3</th>
<th>In the MSU program version</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISSIM</td>
<td>DATA</td>
</tr>
<tr>
<td>DOTHER</td>
<td>LBLOCK</td>
</tr>
</tbody>
</table>

The subroutines are:

- **FIT** Uses latest gradient to get new configuration.
- **SORT** Sorts an array, dragging along one or more other arrays.
- **NEWSTP** Computes new value of STEP, COSAV, ACSAV, etc.
- **PRINDIST** Prints inter-point distances of final configuration.
- **KPACK** Routines for packing and unpacking IJ (see Ref. 3)
- **COMP** Routines to take powers and roots. Negative numbers are complemented, the root or power is taken, and they are recomplemented. Zero and first powers and roots do not use POWRF.
- **CCACT** Routine to convert alphanumerical parameter options into numeric parameters and to set switches based on these options.
- **CCIO** Routine to do numeric conversions for CCACT.

COMP, KPACK, and CCACT are COMPASS routines; the rest are in 3600 FORTRAN.

**Description of Input**

The input data deck should contain the following, in the order listed.

For each data set:
- Option Cards, if any.
- Configuration Deck, if desired.
- Data Description Deck.
- Data.

*continued*
After all the data sets (there may be more than one data set in a run), a STOP card should follow.

Option Cards

Any of the following options may be used. They may be entered in any order. Wherever a normal case is mentioned, it refers to the parameter values the program will assume if the particular option card is omitted. All of these options may be punched free-field, but there should not be spaces between the letters of a single word, or the digits of a single number. More than one option can be punched on a single card, but each word or number must be separated from the next word or number by a blank column. In any case, only the first 72 columns of the card can be used for specifying an option, and each option must be completely specified on one card, i.e., no continuations to a new card are allowed.

(a) The normal case assumes that the entire matrix of similarities (or dissimilarities) is being input. If only the lower half of the matrix (with or without the major diagonal) is being input, punch the word HALFMATRIX on an option card. Obviously, the program will then assume that the data matrix is symmetric.

(b) If the HALFMATRIX option is chosen, the normal case would be for the major diagonal to be included. If the major diagonal is not included, punch the phrase DIAGONAL ABSENT on one of the option cards.

(c) Missing data values are indicated by filling the appropriate cell of the matrix with a value that is less than the cutoff value, which is normally zero. If a different cutoff is desired, punch the word CUTOFF and a real number on one of the option cards, e.g., CUTOFF 3.0.

(d) Scaling is done using the Minkowski r-metric with a normal case value of r=2.0, giving Euclidean distance. If a different value of r is desired, punch the letter R follow by a real number to specify the value of r that is desired, e.g., R 1.0 will give "city-block" distances.

(e) In the normal case, equal data values corresponding to unequal distances in the configuration do not contribute to the stress, and no attempt is made to equalize these distances. This is what Kruscal describes as the "primary" approach. If the "secondary" approach is desired, i.e., if it is desired that unequal distances which correspond to equal data values contribute to the stress, and hence, that an attempt be made to equalize these distances, the word SECONDARY should be punched on an option card.

continued
(f). In the normal case, scaling is done for two dimensions only. Scaling may be done in any number of dimensions from one to ten. In a non-normal case, the maximum number of dimensions desired can be specified by punching the word DIMMAX followed by an integer specifying the maximum number of dimensions desired (up to ten). Similarly, the minimum number of dimensions can be changed from the normal case (two) to some other value by punching the word DIMMIN followed by an integer specifying the minimum number of dimensions desired. In the normal case, a configuration is computed for every number of dimensions from DIMMAX through DIMMIN. If this is not desired, punch the word DIMDIF followed by an integer on an option card. The scaling will then start with DIMMAX dimensions and proceed down by steps of DIMDIF dimensions until DIMMIN is passed. Thus the normal case is equivalent to the following option card:

DIMMAX 2  DIMMIN 2  DIMDIF 1. The option card DIMMAX 10  DIMMIN 2  DIMDIF 3 would cause scaling in 10, 7, and 4 dimensions; note that if DIMMIN had instead been 1, scaling would have been done in 10, 7, 4, and 1 dimensions. Only one or two of the normal case values need be changed; e.g., DIMMAX 10 alone would produce scaling in 10, 9, 8, 7, 6, 5, 4, 3, and 2 dimensions.

(g). Normally, scaling for a given number of dimensions stops if STRESS is less than 0.05, if SFGR is less than or equal to zero, if SRAT is greater than 0.999, or if there have been more than 25 iterations of the scaling routine. A different lower bound can be specified for STRESS by punching the word STRMIN followed by a real number, to specify the requested boundary value, on an option card. A lower bound for SFGR can similarly be specified by punching SFGRMN followed by a real number on an option card. In the same manner an upper bound for SRAT can be specified by punching SRATST followed by a real number and for the number of iterations by punching ITERATIONS followed by an integer.

(h). If it is desired that the final configuration for each number of dimensions be punched onto cards as well as being printed, the word PUNCH should be punched on an option card.

(i). If it is desired that the interpoint distances of the final configuration be printed, the word DISTANCES should be punched on an option card.

Configuration Deck (optional)

Normally, the computer arbitrarily creates a starting configuration for the largest number of dimensions in which scaling has been requested. The results of scaling in each number of
dimensions are then used as a starting configuration for the next smaller number of dimensions in which scaling has been requested. There is, however, a provision for the specification by the user of a starting configuration. A starting configuration can be used to get more iterations on a previously done scaling job, or more important, to specify a different configuration when the arbitrary configuration results in only locally minimum stress. The configuration specified by the user need not have the same dimensionality as the maximum number of dimensions specified. A useful starting configuration is the principal-axis factor solution to the matrix of similarities. A configuration deck is made up in the following manner:

Card 1: the word CONFIGURATION anywhere in Col. 1-72 on a card.

Card 2: in Col. 1-72, a title for the configuration—any BCD characters may be used.

Card 3:  
<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-3</td>
<td>the number of points (the number of rows) of the configuration matrix</td>
</tr>
<tr>
<td>4-6</td>
<td>the number of dimensions (the number of columns) of the configuration matrix</td>
</tr>
</tbody>
</table>

Card 4:  
<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-72</td>
<td>a standard FORTRAN data format for one row of the configuration matrix (This is an F-type format statement; for more information, see Ref. 2.)</td>
</tr>
</tbody>
</table>

Matrix deck: the actual configuration matrix, row by row.

Data Description Cards  
Card 1:  
<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-72</td>
<td>punch SIMILARITIES if the data are a matrix of similarities; DISSIMILARITIES if the data are a matrix of dissimilarities.</td>
</tr>
</tbody>
</table>

Card 2:  
<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-72</td>
<td>the title of the data matrix; any BCD characters may be used.</td>
</tr>
</tbody>
</table>
Card 3: **Columns** 1-3

**Contents**

the number of objects to be scaled (the order of the data matrix).

Card 4: **Columns** 1-72

**Contents**

a FORTRAN F-type format for the longest row of the data matrix.

(For more information, see Ref. 2.)

Data Deck

The data matrix should be inserted at this point. Its form will depend on whether the whole or half matrix is entered.

If the whole matrix is entered, it should be entered one row at a time, with each row starting on a new card. If the half matrix is entered and the diagonal is present, then each row should include only the values up to and including the diagonal. If the diagonal is absent, the data matrix starts with row two of the similarity/dissimilarity matrix, and each row includes the values up to but not including the diagonal element.

Compute Card

After the data matrix, the data deck should contain a card with the word COMPUTE punched somewhere in Col. 1-72.

Stop Card

The last card of the job deck should have punched on it the word STOP somewhere in Col. 1-72.

REFERENCES


SAMPLE INPUT
'JOB,015525,KRUSCAL,3.0,APHL
'EQUIP,17=(CISSL BINARY),MT(537),RO
'LOAD,17
'RUN,5.00,2000.
$ROUTINE,KRUSKAL
'MARKEF,69
'LOAD
'RUN,5.00,2000.
DIMMAX 10
DIMMIN 1
HALFMATRIX
CUTOFF -.05
SECONDARY
ITERAT 100
DISSIMULAR
DISTANCES OF REGULAR HEXAGON
6
(6F10.9)
0000000000
10000000000000000000000000
173205000010000000000000000000000000
200000000173205000100000000000000000000000000
173205000200000000173205000010000000000000000000000000
100000000173205000020000000173205000010000000000000000000000000

COMPUTE
STOP
SAMPLE OUTPUT

DIMMAX 10
DIMMIN 0
HALFMATRIX = 1
CUTOFF = 0
SECONDARY = 0
ITERAT 100

DISTANCES OF REGULAR HEXAGON
a = 0

BECAUSE THE NUMBER OF REPLICATES WAS NOT SPECIFIED, IT IS ASSUMED TO BE 1.

COMPUTE = 0

HISTORY OF COMPUTATION, N = 6, DIMENSION = 10

ITERATION STRESS  SRAT  SRTAV  CARG  CAG4  ACSAV  SFGT  STEP

1 0.273 0.000 0.000 0.000 0.000 0.113 1.5550

2 1.791 1.033 0.480 -0.025 -0.017 0.017 0.009 1.3892

3 0.275 1.303 1.003 -0.400 -0.260 0.289 0.127 0.6693

4 0.194 0.513 0.652 -0.390 -0.359 0.355 0.096 0.3083

5 0.070 0.342 0.614 -0.190 -0.251 0.251 0.101 0.1884

6 0.056 0.706 0.470 -0.065 -0.056 0.056 0.112 0.0746

7 0.014 0.245 0.470 0.036 -0.001 0.445 0.036 0.0349

continued
ITERATION STRESS SRAT SRATAV CAGRGL COSAV ACSAV SFGR STEP

SATISFACTORY STRESS WAS REACHED

THE FINAL CONFIGURATION OF 6 POINTS IN 10 DIMENSIONS HAS STRESS 0.006

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.534</td>
<td>-0.473</td>
<td>-0.410</td>
<td>-0.399</td>
<td>-0.391</td>
<td>0.080</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>0.255</td>
<td>0.103</td>
<td>0.381</td>
<td>-0.214</td>
<td>-0.234</td>
<td>0.222</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>3</td>
<td>0.160</td>
<td>-0.115</td>
<td>0.280</td>
<td>-0.219</td>
<td>-0.279</td>
<td>0.220</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>4</td>
<td>0.077</td>
<td>-0.226</td>
<td>-0.111</td>
<td>-0.153</td>
<td>-0.144</td>
<td>0.171</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>5</td>
<td>0.036</td>
<td>0.192</td>
<td>0.072</td>
<td>-0.064</td>
<td>0.108</td>
<td>-0.969</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>6</td>
<td>0.036</td>
<td>0.192</td>
<td>0.072</td>
<td>-0.064</td>
<td>0.108</td>
<td>-0.969</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

*********

HISTORY OF COMPUTATION, N = 6, DIMENSION = 9

ITERATION STRESS SRAT SRATAV CAGRGL COSAV ACSAV SFGR STEP

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.534</td>
<td>-0.473</td>
<td>-0.410</td>
<td>-0.399</td>
<td>-0.391</td>
<td>0.080</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>0.255</td>
<td>0.103</td>
<td>0.381</td>
<td>-0.214</td>
<td>-0.234</td>
<td>0.222</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>3</td>
<td>0.160</td>
<td>-0.115</td>
<td>0.280</td>
<td>-0.219</td>
<td>-0.279</td>
<td>0.220</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>4</td>
<td>0.077</td>
<td>-0.226</td>
<td>-0.111</td>
<td>-0.153</td>
<td>-0.144</td>
<td>0.171</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>5</td>
<td>0.036</td>
<td>0.192</td>
<td>0.072</td>
<td>-0.064</td>
<td>0.108</td>
<td>-0.969</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>6</td>
<td>0.036</td>
<td>0.192</td>
<td>0.072</td>
<td>-0.064</td>
<td>0.108</td>
<td>-0.969</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

*********

HISTORY OF COMPUTATION, N = 6, DIMENSION = 8

*********

HISTORY OF COMPUTATION, N = 6, DIMENSION = 5

ITERATION STRESS SRAT SRATAV CAGRGL COSAV ACSAV SFGR STEP

D=DA
D=DA
D=DA
D=DA
D=DA
D=DA
D=DA

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.185</td>
<td>0.800</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.0329</td>
<td>0.5547</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.028</td>
<td>0.101</td>
<td>0.762</td>
<td>0.037</td>
<td>0.025</td>
<td>0.025</td>
<td>0.0985</td>
<td>0.2919</td>
</tr>
</tbody>
</table>

continued
ITERATION STRESS  SRAT  SRATAV  CAGRGL  COSAV  ACSAV  SFOR  STEP
SATISFACTORY STRESS WAS REACHED

THE FINAL CONFIGURATION OF 6 POINTS IN 5 DIMENSIONS HAS STRESS 0.003

FINAL CONFIGURATION
1  2  3  4  5
1 0.557 -0.459 -0.411 -0.399 -0.380
2 -0.191 0.637 -0.458 -0.179 -0.319
3 -0.352 -0.247 -0.611 -0.226 -0.241
4 -0.042 -0.207 -0.098 0.961 -0.130
5 -0.030 -0.346 -0.106 -0.164 0.095
6 0.638 0.618 0.335 0.022 0.324

**********

HISTORY OF COMPUTATION, N=6, DIMENSION=4

ITERATION STRESS  SRAT  SRATAV  CAGRGL  COSAV  ACSAV  SFOR  STEP
D=0A  D=0A  D=0A  D=0A  D=0A  D=0A
0 0.194 0.880 0.200 0.000 0.000 0.0936 0.4118

D=0A  D=0A  D=0A  D=0A  D=0A  D=0A  D=0A
19 0.104 1.000 0.999 0.926 0.900 0.869 0.0013 0.0021

MINIMUM WAS ACHIEVED

continued
The final configuration of 6 points in 4 dimensions has stress 0.104

**Final Configuration**

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>0.714</td>
<td>-0.437</td>
<td>-0.387</td>
</tr>
<tr>
<td>2</td>
<td>-0.495</td>
<td>0.751</td>
<td>-0.446</td>
</tr>
<tr>
<td>3</td>
<td>-0.271</td>
<td>0.173</td>
<td>0.937</td>
</tr>
<tr>
<td>4</td>
<td>0.208</td>
<td>-0.185</td>
<td>-0.391</td>
</tr>
<tr>
<td>5</td>
<td>-0.646</td>
<td>-0.627</td>
<td>-0.370</td>
</tr>
<tr>
<td>6</td>
<td>0.645</td>
<td>0.646</td>
<td>0.357</td>
</tr>
</tbody>
</table>

************

**History of Computation, N= 6, Dimensions= 4**

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Stress</th>
<th>SRAT</th>
<th>SRATAV</th>
<th>CAGRGL</th>
<th>COSAV</th>
<th>ACASV</th>
<th>SFGR</th>
<th>STEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>DaOA</td>
<td>0.024</td>
<td>0.600</td>
<td>-0.000</td>
<td>-0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.220</td>
<td>0.220</td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Satisfactory stress was reached.

The final configuration of 6 points in 3 dimensions has stress 3.010

**Final Configuration**

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>0.445</td>
<td>0.702</td>
<td>-0.595</td>
</tr>
<tr>
<td>2</td>
<td>-0.468</td>
<td>0.875</td>
<td>-0.135</td>
</tr>
<tr>
<td>3</td>
<td>-0.672</td>
<td>-0.277</td>
<td>0.488</td>
</tr>
<tr>
<td>4</td>
<td>-0.445</td>
<td>-0.706</td>
<td>0.547</td>
</tr>
<tr>
<td>5</td>
<td>0.229</td>
<td>-0.998</td>
<td>-0.170</td>
</tr>
<tr>
<td>6</td>
<td>0.911</td>
<td>-0.198</td>
<td>-0.870</td>
</tr>
</tbody>
</table>

************

**History of Computation, N= 6, Dimensions= 3**

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Stress</th>
<th>SRAT</th>
<th>SRATAV</th>
<th>CAGRGL</th>
<th>COSAV</th>
<th>ACASV</th>
<th>SFGR</th>
<th>STEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>DaOA</td>
<td>0.117</td>
<td>0.800</td>
<td>0.900</td>
<td>0.900</td>
<td>0.000</td>
<td>0.184</td>
<td>0.638</td>
<td>0.638</td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**History of Computation, N= 6, Dimensions= 2**

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Stress</th>
<th>SRAT</th>
<th>SRATAV</th>
<th>CAGRGL</th>
<th>COSAV</th>
<th>ACASV</th>
<th>SFGR</th>
<th>STEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>DaOA</td>
<td>0.024</td>
<td>0.600</td>
<td>0.900</td>
<td>0.900</td>
<td>0.000</td>
<td>0.184</td>
<td>0.638</td>
<td>0.638</td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Iteration stress reached.

The final configuration of 6 points in 2 dimensions has stress 0.000

**Final Configuration**

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>0.117</td>
<td>0.800</td>
<td>0.900</td>
</tr>
<tr>
<td>2</td>
<td>0.911</td>
<td>-0.198</td>
<td>-0.870</td>
</tr>
</tbody>
</table>

**History of Computation, N= 6, Dimensions= 1**

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Stress</th>
<th>SRAT</th>
<th>SRATAV</th>
<th>CAGRGL</th>
<th>COSAV</th>
<th>ACASV</th>
<th>SFGR</th>
<th>STEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>DaOA</td>
<td>0.010</td>
<td>0.600</td>
<td>0.900</td>
<td>0.900</td>
<td>0.000</td>
<td>0.184</td>
<td>0.638</td>
<td>0.638</td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DaOA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SATISFACTORY STRESS WAS REACHED.

THE FINAL CONFIGURATION OF 6 POINTS IN 2 DIMENSIONS HAS STRESS 0.606.

FINAL CONFIGURATION

1 1 0.643 0.772
2 0.634 0.939
3 0.629 0.766
4 0.353 0.924
5 0.982 0.199

********

HISTORY OF COMPUTATION, N = 6, DIMENSION 1

ITERATION STRESS SRAT SRATAV CARCHG COSAV ACSAV SFGR STEP

D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA

D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA

D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA

D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA

D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA

D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA
D#DA

14 0.413 1.000 0.999 -0.326 -0.033 0.594 0.0001 0.0021

15 0.413 1.000 0.999 -0.997 -0.656 0.846 0.0006 0.0005

16 0.413 1.000 0.999 1.000 0.432 0.948 0.0004 0.0004

MINIMUM WAS ACHIEVED.

THE FINAL CONFIGURATION OF 6 POINTS IN 1 DIMENSIONS HAS STRESS 0.413.

FINAL CONFIGURATION

1 1.364
2 1.001
3 0.366
4 1.001
5 1.366
6 0.366

********

***************
COST ESTIMATE

For the job listed on the Sample Output, the total processing time was 88.912 seconds. At the current rate for the Michigan State University's CDC 3600 ($245./hr.), the computer time cost $6.05 plus a charge for input and output. A 10% surcharge will be assessed if the billing is to a non-University account. In addition, there is a consulting fee of $10./hr. for work done by the Applications Programming Group.

Charge to user = computer costs + consulting + network overhead
= $6.10 (approx.) + consulting + network overhead

CONTENTS—SUBROUTINE KRUSCAL

<table>
<thead>
<tr>
<th>pages</th>
<th>Identification &amp; Abstract</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td></td>
</tr>
<tr>
<td>3-8</td>
<td>User Instructions</td>
</tr>
<tr>
<td>9-14</td>
<td>I/O</td>
</tr>
<tr>
<td>15</td>
<td>Cost—Contents</td>
</tr>
</tbody>
</table>