NOTICE: When government or other drawings, specifications or other data are used for any purpose other than in connection with a definitely related government procurement operation, the U. S. Government thereby incurs no responsibility, nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.
Business Management Game: Part 10

Instructions for the Use and Modification of Program UMPIRE
TECHNICAL MEMORANDUM
(TM Series)

This document was produced by SDC in performance of U.S. Government Contracts

Business Management Game, Part III:
Instructions for the Use and Modification
of Program UMPIRE

Sandra Peterson
May 20, 1963

The views, conclusions or recommendations expressed in this document do not necessarily reflect the official views or policies of agencies of the United States Government.

Permission to quote from this document or to reproduce it, wholly or in part, should be obtained in advance from the System Development Corporation.
PREFACE

This document describes the use, modification and maintenance of program UMPIRE, a program written in JOVIAL for the Philco 2000, to be used in the play of the management game described in TM-1088. The program can be used to replace some of the manual operations involved in umpiring for this game, which is based on the Andlinger-Green game described in the Harvard Business Review for March and April, 1958 by G. R. Andlinger.

While the document describes in some detail those umpiring functions which are unique to the use of the program, it relies heavily on TM-1088 and TM-1088/001/00 for precise descriptions of those functions which are common to players and umpires or to umpires with and without the program. Since the program does not deal with the financial statements, those aspects of umpiring which are concerned with them are not covered.

The author wishes to thank Richard Gilinsky and Patricia Kenney for their constant assistance and advice in the writing and checkout of program UMPIRE.

Sandra Peterson
1. General Information ........................................... 5
   1.1 Responsibilities ....................................... 5
   1.2 Functions ............................................. 5
       1.2.1 Market ........................................ 5
       1.2.2 Sales ........................................ 5
       1.2.3 Consulting ..................................... 5
       1.2.4 Communication with the Computer ............... 6
   1.3 Allocation of Functions ............................... 6
   1.4 Restrictions ......................................... 6
       1.4.1 Number of Teams ................................ 6
       1.4.2 Copies of Quarterly Report ..................... 6
       1.4.3 Sales ......................................... 6

2. Starting Position ........................................... 6
   2.1 New Corporation ...................................... 7
   2.2 New Management of Old Corporation ................... 7

3. Market .................................................................. 7

4. Sales ................................................................... 7

5. Consulting ....................................................... 7

6. Communication with the Computer .......................... 7
   6.1 Loading the Program ................................... 8
   6.2 Inputs ................................................... 8
       6.2.1 General Rules .................................. 8
       6.2.2 Number of Teams ................................ 8
       6.2.3 Quarter Number .................................. 8
       6.2.4 Units Scheduled .................................. 9
       6.2.5 Factoring ......................................... 9
       6.2.6 Construction ...................................... 10
       6.2.7 Hiring ............................................ 10
       6.2.8 R and D .......................................... 10
       6.2.9 Consulting ....................................... 10
       6.2.10 Advertising ..................................... 10
       6.2.11 Sales ........................................... 11
       6.2.12 Product Improvement ......................... 11
       6.2.13 Production Lines Junked ....................... 12
6.3 Outputs ........................................................................................................ 12
   6.3.1 Sales Error ............................................................................................ 12
   6.3.2 Quarterly Report .................................................................................... 12
6.4 Restarting Umpire ....................................................................................... 12

7. Game Boards ................................................................................................. 13

8. Equipment ..................................................................................................... 13

9. Sequence ........................................................................................................ 13

Appendices

1. Umpire Messages ........................................................................................... 15
2. Umpire Quarterly Report .............................................................................. 21
3. Modifications to Program Umpire ............................................................... 27
4. Program Listings ............................................................................................ 33
   Quarter 1 ....................................................................................................... 39
   Quarter 20 .................................................................................................... 52
BUSINESS MANAGEMENT GAME, PART III:
INSTRUCTIONS FOR THE USE AND MODIFICATION OF PROGRAM UMPIRE

1. GENERAL

1.1 RESPONSIBILITIES

The responsibilities of the umpires when the automated umpiring function is used are somewhat different from their responsibilities when the umpiring function is entirely manual. The umpires, should, however, acquaint themselves with the responsibilities of the umpires in the manual circumstance.

1.2 FUNCTIONS

The functions of the umpires in the automated umpiring mode of play encompass three of the six functions listed in volume two of this document, with one additional function.

1.2.1 Market

The following functions which deal with the market must be performed manually by the umpires:

a. Arraying the market initially.
b. Decreasing the market as a result of sales.
c. Updating the market for the next quarter.

1.2.2 Sales

The following functions with respect to sales must be performed by the umpires:

a. Determining product improvement on the basis of R and D investment.
b. Determining sales probability based on advertising and product improvement.
c. Determining sales.
d. Splitting sales if more than one company sells to a customer.

1.2.3 Consulting

The umpires must provide consulting services for the players when they communicate their decision to purchase it by indicating its cost on the Quarterly Report.
1.2.4 Communication with the Computer

The umpires are responsible for communicating player and umpire decisions to the computer for each team. The computer program UMPIRE will then produce correct Quarterly Reports to be returned to the Board of Directors before the next quarter begins.

1.3 ALLOCATION OF FUNCTIONS

The various functions performed by the umpires may be allocated in any convenient way. If the umpires are located at any distance from the computer, however, it is advisable to locate an umpire at the computer to receive inputs by telephone. It is estimated that one umpire will be needed (aside from the umpire at the computer) for four teams or fewer; if from 5 to 7 teams are playing, two umpires will be needed.

1.4 RESTRICTIONS

The computer program UMPIRE imposes restrictions on three areas of the game. The restrictions can be changed easily by slight program modifications. (See Appendix III).

1.4.1 Number of Teams

The program currently provides for a maximum of seven teams playing the game.

1.4.2 Copies of Quarterly Report

The line printer which is used by program UMPIRE to produce copies of the correct Quarterly Report to be returned to the players will handle two types of paper: no carbon, or one carbon (the latter is referred to as "two part" paper). If no carbon is used, three copies of the Quarterly Report for each team will be produced. If carbon is used, six copies will be produced.

1.4.3 Sales

The program currently provides that one team may make sales to no more than 10 customers in any one quarter.

2. STARTING POSITION

Copies of the program UMPIRE are available to provide for two starting positions at the beginning of play. It is possible to modify the program to begin at other points. (See Appendix III).
2.1 NEW CORPORATION

The game begins with all corporations just starting in business. To play the game with this beginning position, use UMPIRE deck 1.

2.2 NEW MANAGEMENT OF OLD CORPORATION

The game begins with each Board of Directors assuming control of a corporation which has been operating for some length of time. Deck 2 of program UMPIRE is set up to begin at quarter 20, according to the history and starting Quarterly Report shown in Appendix I and Appendix II, respectively, of volume one of this document. The game board at the beginning of play is illustrated in Appendix III of volume one of this document. If this starting position is to be used, the umpires should consult section 2.2 of volume one to insure that the players are given sufficient beginning information.

3. MARKET

The activities which the umpires must perform in conjunction with the market are identical to those which must be performed in the case of completely manual umpiring. The description of these activities and the explanation of the methods can be found in section 3 of TM-1088/001/00.

4. SALES

The functions of the umpires in the area of sales are identical to those performed in the completely manual method of umpiring. Instructions for these functions can be found in section 5 of TM-1088/001/00.

5. CONSULTING

The consulting services which the umpires must provide in this umpiring mode are the same as those performed in the manual umpiring mode, and can be found in section 7 of volume one of this document.

6. COMMUNICATION WITH THE COMPUTER

Program UMPIRE is designed to operate on the Philco 2000 computer, located in the SSRL facility in Santa Monica. It assumes responsibility for all of the umpiring functions in the areas of Costs and Computations, as well as the responsibility for determining salesman losses. There are several umpire responsibilities which must be performed in order for the program to perform its duties correctly.
6.1 LOADING THE PROGRAM

When the umpires are ready for the program to perform its duties for the first time, they must load the program into the core memory of the computer. The umpire need only present the correct deck of UMPIRE (deck 1 for new corporation, deck 2 for quarter 20) to the computer operator to be loaded. The program will begin operating once it is loaded.

6.2 INPUTS

The program will ask certain questions via the typewriter at the computer console. The umpires should provide answers to the questions on the typewriter. All answers will be numbers. A sample of the typewriter messages and responses may be found in Appendix I.

6.2.1 General Rules

Each time the program asks a question, it will type out the question and provide a carriage return and shift to upper case. In providing his response, the umpire should do the following.

a. Type the correct number and hit Carriage Return.

b. If an error is made, hit the Stop Code button. Begin the response again. If an error is noticed after the Carriage Return, it is impossible to recover. Because of this, care should be used in entering values.

6.2.2 Number of Teams

In the first quarter to be played, the first question to be asked by the program is "HOW MANY TEAMS ARE PLAYING." Respond with a one digit number from 1 to 7 and hit Carriage Return. This question will be asked only once.

6.2.3 Quarter Number

At the beginning of each quarterly running of the umpiring program, the following two lines will be typed:

"PLEASE ENTER THE FOLLOWING VALUES"
"QUARTER NUMBER"

The umpire should respond with the correct quarter number, and a Carriage Return. If play begins with a new corporation, the first quarter number should be 1. If deck 2 is used, the first quarter number should be 20.
6.2.4 Units Scheduled

After the Quarter Number has been entered, the program will begin to ask questions for each of the teams in sequence, from team 1 to the largest team number. All questions will be asked for team 1, then all questions for team 2, etc.

As UMPIRE begins the processing for each team, it will type the following lines:
"FOR TEAM ___" (the team number will be filled in)
"UNITS SCHEDULED"

The umpire should respond with the number entered by the players on the "Units Scheduled" line on page one of the Quarterly Report. If no number has been entered on the Quarterly Report, enter zero. Follow the entry with a Carriage Return.

6.2.5 Factoring

The next questions asked by UMPIRE for the team currently being processed deal with "Current Factoring" specifications as found on the Quarterly Report. The following lines will be typed:

a. "BLOCK 2 FACTORING" The response must be the "Cash to be Realized" item listed on the "A/R Block 2 @ 90%" line of the Quarterly Report form. It is important to note that the response must be in dollars to be realized, not in number of units factored. Carriage Return will cause the next question to be typed.

b. "BLOCK 3" The response must be the "Cash to be Realized" item on the "A/R Block 3 @ 90%" line on the Quarterly Report form, followed by a Carriage Return.

c. "BLOCK 4" The response must be the "Cash to be Realized" item on the "A/R Block 4 @ 80%" line on the Quarterly Report form, followed by a Carriage Return.

d. "BLOCK 5" The response must be the "Cash to be Realized" item on the "A/R Block 5 @ 80%" line on the Quarterly Report form, followed by a Carriage Return.

In all cases, if no entry has been made on the Quarterly Report form in the position specified, enter zero and hit Carriage Return.
6.2.6 Construction

The next line to be typed by UMPIRE is as follows: "CONSTRUCTION." The umpire should respond with the total cost of construction in this quarter for the team being processed. The value can be found on the "Construction" line of the Quarterly Report form. Enter the value and Carriage Return.

6.2.7 Hiring

The next inquiry is made with respect to the hiring decisions made by the team members. The line typed is: "HIRING." The umpires should respond with the total cost of the hiring done by the team. The information can be found on the "Hiring" line under Disbursements on the Quarterly Report form. Enter the value and hit Carriage Return.

6.2.8 R and D

UMPIRE next requests the expenditures of the corporation in this quarter on R and D. The line typed by UMPIRE is: "R AND D." The umpire should respond with the value found on the "R & D" line under Disbursements on the Quarterly Report and a Carriage Return.

6.2.9 Consulting

The information requested by UMPIRE at this point is the total cost of consulting services requested in the current quarter by the team being processed. The umpire has the responsibility of checking the computation of total consulting costs. When the program types "CONSULTING" respond with the total consulting fees found on the "Consulting Fees" line under Disbursements on the Quarterly Report and a Carriage Return.

6.2.10 Advertising

UMPIRE will next type "ADS." The umpire must respond with the total pages of ads found to the left of the "Pages of Ads" line under Disbursements on the Quarterly Report form. It is important to note that pages, and not dollars, is the unit of the response to this query. Follow the response with a Carriage Return.
6.2.11 Sales

In the area of sales the program will, under certain circumstances, make more than one request for information. The following lines are typed by UMPIRE:

a. "SALES" The umpire should respond with the total sales made by the team in the current quarter. The information can be found on page two of the Quarterly Report, on the "Units Sold" line. Hit Carriage Return.

b. If the response made to "SALES" was not zero, UMPIRE will type the following message: "CUSTOMERS AND UNITS SOLD." The umpire is expected to respond by:

1) Entering the number of the customer to whom a sale was made followed by a Carriage Return.

2) Entering the number of units sold to this customer by the company being processed followed by a Carriage Return.

3) Repeating 1) and 2) until all customers to whom sales were made by the company being processed and the number of units sold to them have been listed.

4) If sales were made to 10 customers by the company, the sequence will automatically end and UMPIRE will go to its next message.

5) If fewer than 10 customers were listed, the umpire must terminate the sequence by entering customer number zero and hitting Carriage Return.

6) UMPIRE will not accept more than 10 customers and units sold for any one team.

6.2.12 Product Improvement

The next line typed by UMPIRE reads: "PRODUCT IMPROVEMENT, 1 OR 0 " The umpire should respond by entering 1 and Carriage Return if the team achieved a product improvement in the current quarter, or 0 and Carriage Return if no product improvement was achieved.
6.2.13 **Production Lines Junked**

The next message typed by **UMPIRE** is:

"PROD LINES JUNKED." The umpire is expected to respond with the number of production lines junked by the team during the current quarter. This information can be found on page one of the Quarterly Report on the "Production Line Scraped" line. If no entry was made by the team, enter zero. The umpire must provide a Carriage Return.

6.3 **OUTPUTS**

The program **UMPIRE** will provide outputs of two kinds.

6.3.1 **Sales Error**

Since **UMPIRE** keeps records of the number of units each team has in inventory, it can recognize the fact that the sales assigned to a team exceed the inventory of that team. If this occurs, **UMPIRE** will reduce the total sales to the number of units in inventory, but will not change the listing of the customers and the units sold to them. The umpire must correct this item himself. **UMPIRE** will warn the umpire located at the console typewriter that this error has occurred by typing the following message before the next team is processed: "SALES HAVE BEEN REDUCED TO THE NUMBER OF UNITS IN INVENTORY."

6.3.2 **Quarterly Report**

**Program UMPIRE** produces a quarterly report for each team. (See Section 1.4.2). After processing all teams for one quarter, **UMPIRE** will stop. At this time, the computer operators should be instructed to print the PRINT tape. The output will be the quarterly reports for all the teams playing. A sample of the Quarterly Report printed by **UMPIRE** will be found in Appendix II.

6.4 **RESTARTING UMPIRE**

In order to restart **UMPIRE** for the next quarter, the umpire need only instruct the computer operator to "advance" the computer. **UMPIRE** will continue with the messages "PLEASE ENTER THE FOLLOWING VALUES" and "QUARTER NUMBER." It is important to note that the computer must remain idle until subsequent quarters are to be played. Due to its responsibilities for "remembering," **UMPIRE** may not be removed from memory between quarters of play.
7. **GAME BOARDS**

The umpires are responsible for keeping their game boards for the teams updated according to the decisions made by the team members, the umpires' decisions, and the decisions made by UMPIRE. On receiving copies of the completed Quarterly Report from UMPIRE, the umpires should pay particular attention to the salesman losses, and record these events on the game board.

8. **EQUIPMENT**

The umpires should have the same equipment available to them as described in section 8 of volume one of this document.

9. **SEQUENCE**

The sequence of activities to be followed by the umpires is as listed in this section. References made are to other sections of this document which describe the particular function in greater detail.

<table>
<thead>
<tr>
<th>Umpiring Activities</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>9.1 BEFORE PLAY BEGINS</strong></td>
<td></td>
</tr>
<tr>
<td>9.1.1 Array the market</td>
<td>TM-1088/001/00, Section 3.2</td>
</tr>
<tr>
<td>9.1.2 Set up game boards</td>
<td></td>
</tr>
<tr>
<td>a. Quarter 1 boards are empty</td>
<td>TM-1088/001/00, Appendix TTT</td>
</tr>
<tr>
<td>b. Quarter 20</td>
<td></td>
</tr>
<tr>
<td><strong>9.2 AFTER BOARD MEETING</strong></td>
<td></td>
</tr>
<tr>
<td>9.2.1 Determine accumulated R &amp; D</td>
<td>TM-1088/001/00, Section 5.1.1</td>
</tr>
<tr>
<td>9.2.2 Determine product improvement (5.1.4)</td>
<td>TM-1088/001/00, Section 5.1.2</td>
</tr>
<tr>
<td>9.2.3 Determine sales probabilities</td>
<td>TM-1088/001/00, Section 5.2</td>
</tr>
<tr>
<td>9.2.4 Determine sales</td>
<td>TM-1088/001/00, Section 5.3</td>
</tr>
<tr>
<td>9.2.5 Determine split sales</td>
<td>TM-1088/001/00, Section 5.4</td>
</tr>
<tr>
<td>9.2.6 Document sales</td>
<td>TM-1088/001/00, Section 5.5</td>
</tr>
<tr>
<td>9.2.7 Record team decisions on board</td>
<td>TM-1088/000/00, Section 15</td>
</tr>
<tr>
<td></td>
<td>Task</td>
</tr>
<tr>
<td>---</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>a</td>
<td>Current factoring</td>
</tr>
<tr>
<td>b</td>
<td>Construction</td>
</tr>
<tr>
<td>c</td>
<td>Salesman hire</td>
</tr>
<tr>
<td>d</td>
<td>Units scheduled</td>
</tr>
<tr>
<td>e</td>
<td>Production lines scrapped</td>
</tr>
<tr>
<td>9.2.8</td>
<td>Run program UMPIRE</td>
</tr>
<tr>
<td>9.2.9</td>
<td>Update market for next quarter</td>
</tr>
<tr>
<td>9.2.10</td>
<td>Complete consulting forms</td>
</tr>
<tr>
<td>9.2.11</td>
<td>Receive QRs from UMPIRE</td>
</tr>
<tr>
<td>9.2.12</td>
<td>Update game board</td>
</tr>
<tr>
<td>9.2.13</td>
<td>Give UMPIRE QR, consulting forms and blank QR to players</td>
</tr>
</tbody>
</table>
APPENDIX I

UMPIRE MESSAGES
UMPIRE
HOW MANY TEAMS ARE PLAYING
2

PLEASE ENTER THE FOLLOWING VALUES
QUARTER NUMBER
1

FOR TEAM 1
UNITS SCHEDULED
0

BLOCK 2 FACTORING
0

BLOCK 3
0

BLOCK 4
0

BLOCK 5
0

CONSTRUCTION
180000

HIRING
50000

R AND D
5000

CONSULTING
0

ADS
0

SALES
0

PRODUCT IMPROVEMENT, 1 OR 0
0

PROD LINES JUNKED
0

FOR TEAM 2
UNITS SCHEDULED
0

BLOCK 2 FACTORING
0

BLOCK 3
0

BLOCK 4
0

BLOCK 5
0

CONSTRUCTION
20 May 1963

Note that this entry is an error, and that the corresponding Quarterly Report in Appendix II shows "0" total salesmen.

150000

HIRING
3000

R AND D
5000

CONSULTING
0

ADS
0

SALES
0

PRODUCT IMPROVEMENT, 1 OR 0
0

PROD LINES JUNKED
0
UNPIRE
HOW MANY TEAMS ARE PLAYING?

PLEASE ENTER THE FOLLOWING VALUES

QUARTER NUMBER
20

FOR TEAM 1
UNITS SCHEDULED
7

BLOCK 2 FACTORING
18000

BLOCK 3
27000

BLOCK 4
16000

BLOCK 5
24000

CONSTRUCTION
0

HIRING
0

R AND D
5000

CONSULTING
2000

ADS
5

SALES
10

CUSTOMERS AND UNITS SOLD
2

3
5
4
16
3
0
PRODUCT IMPROVEMENT, 1 OR 0
1

PROD LINES JUNKED
0

FOR TEAM 2
UNITS SCHEDULED
10

BLOCK 2 FACTORING
0

BLOCK 3
0

BLOCK 4
80000

BLOCK 5
0

CONSTRUCTION
0

HIRING
10000

R AND D
0

CONSULTING
0

ADS
3

SALES
9

CUSTOMERS AND UNITS SOLD
4

2

6

1

7

1

10

1

12

2

21

2

0
PRODUCT IMPROVEMENT, 1 or 0

PROD LINES JUNKED
0
APPENDIX II

UMPIRE QUARTERLY REPORT
<table>
<thead>
<tr>
<th>QUANTITY REPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEM NUMBER 1</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>INVENTORY</th>
<th>SALES</th>
<th>R AND D</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEGINNING CASH</td>
<td>24,960</td>
<td></td>
</tr>
<tr>
<td>CUSTOMER UNITS SHIPPED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUMULATIVE N AND I</td>
<td>9,000</td>
<td></td>
</tr>
<tr>
<td>PRODUCT INVENTION NOT ACHIEVED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VENDOR AR TO PAH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL FACTORING</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>TOTAL AVAILABLE CASH</td>
<td>40,000</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QUANTITY FACTORING</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT 90 PERCENT BLOCK 2</td>
</tr>
<tr>
<td>BLOCK 3</td>
</tr>
<tr>
<td>AT 80 PERCENT BLOCK 4</td>
</tr>
<tr>
<td>BLOCK 5</td>
</tr>
<tr>
<td>TOTAL FACTORING A/N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DISPOSALS</th>
<th>TOTAL SALES</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIXED COST</td>
<td>PRODUCTION</td>
<td>0</td>
</tr>
<tr>
<td>VARIABLE COST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SALES SALARIES</td>
<td>UNITS IN INVENTORY</td>
<td>0</td>
</tr>
<tr>
<td>CONSTRUCTION COST</td>
<td>UNITS IN IP</td>
<td>0</td>
</tr>
<tr>
<td>SALESMAN WAGE</td>
<td>PERSONNEL</td>
<td>0</td>
</tr>
<tr>
<td>M AND O EXPENSE</td>
<td>SALESMAN COST</td>
<td>0</td>
</tr>
<tr>
<td>ADVERTISING EXPENSE</td>
<td>IN THE FUND</td>
<td>0</td>
</tr>
<tr>
<td>CONSULTING FEE</td>
<td>IN TRAINING BLOCK 2</td>
<td>0</td>
</tr>
<tr>
<td>TOTAL DISPOSALS</td>
<td>BLOCK 3</td>
<td>0</td>
</tr>
<tr>
<td>BEGINNING CASH</td>
<td>16,900</td>
<td></td>
</tr>
<tr>
<td>CASH LESS DISPOSALS</td>
<td>BLOCK 4</td>
<td>0</td>
</tr>
<tr>
<td>MATURED A/R AT PAH</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>ESTATE ESTATE</td>
<td>16,500</td>
<td></td>
</tr>
<tr>
<td>IN HORIZON</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>FROM FACTORING</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>PLANT VALUE</td>
<td>19,000</td>
<td></td>
</tr>
<tr>
<td>TOTAL SALES</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>TOTAL AVAILABLE CASH</td>
<td>16,900</td>
<td></td>
</tr>
<tr>
<td>Item Description</td>
<td>Quantity/Amount</td>
<td></td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>--------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Beginning Cash</td>
<td>400000</td>
<td></td>
</tr>
<tr>
<td>Matured A/P at Past Factoring</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Total Available Cash</td>
<td>400000</td>
<td></td>
</tr>
<tr>
<td>Current Factoring</td>
<td></td>
<td></td>
</tr>
<tr>
<td>At 60 Percent Block 1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Block 3</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>At 60 Percent Block 4</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Block 5</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Total Factoring A/P</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Disbursements</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Sales</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Fixed Cost</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Variable Cost</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Sales Salaries</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Wages/Compensation</td>
<td>150000</td>
<td></td>
</tr>
<tr>
<td>Supplies/Hire</td>
<td>3000</td>
<td></td>
</tr>
<tr>
<td>H and D Expense</td>
<td>3000</td>
<td></td>
</tr>
<tr>
<td>Salesmen Lost</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Advertising Expense</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Cons Red. Expense</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Total Disbursements</td>
<td>150000</td>
<td></td>
</tr>
<tr>
<td>Block 3</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Beginning Cash</td>
<td>242000</td>
<td></td>
</tr>
<tr>
<td>Cash Less Disbursements</td>
<td>242000</td>
<td></td>
</tr>
<tr>
<td>Total Cash-Flow</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Plant Value</td>
<td>150000</td>
<td></td>
</tr>
<tr>
<td>Total Sales-Flow</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Total Available Cash</td>
<td>242000</td>
<td></td>
</tr>
</tbody>
</table>
## Quarterly Report

**Team Number 1**  
**Quarter Number 26**

<table>
<thead>
<tr>
<th>Cash Items</th>
<th>Sales</th>
<th>R and D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beginning Cash</td>
<td>S</td>
<td>C</td>
</tr>
<tr>
<td>Natural A/R at Pan</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Total Available Cash</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Total Factures A/R</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Current Facturing

- At 90% Percent Block 2: 26240
- Block 3: 209600
- At 60% Percent Block 4: 21710
- Block 5: 209600
- Total Factures A/R: 85000

### Cash Movements

<table>
<thead>
<tr>
<th>Item</th>
<th>Total Sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed Cost</td>
<td>14400</td>
</tr>
<tr>
<td>Variable Cost</td>
<td>19000</td>
</tr>
<tr>
<td>Fixed Salaries</td>
<td>3090</td>
</tr>
<tr>
<td>Cancellation Cost</td>
<td>0</td>
</tr>
<tr>
<td>Salesman Mileage</td>
<td>6</td>
</tr>
<tr>
<td>E and O Expense</td>
<td>3900</td>
</tr>
<tr>
<td>Advertising Expense</td>
<td>13000</td>
</tr>
<tr>
<td>Cancellation Fees</td>
<td>4000</td>
</tr>
<tr>
<td>Total Disbursements</td>
<td>6260</td>
</tr>
<tr>
<td>Cash Less Disbursements</td>
<td>12100</td>
</tr>
<tr>
<td>Plant Value</td>
<td>10000</td>
</tr>
<tr>
<td>Total Salesmen</td>
<td>3</td>
</tr>
</tbody>
</table>

### For Quarter Number 26

<table>
<thead>
<tr>
<th>Item</th>
<th>Total Salesmen</th>
</tr>
</thead>
<tbody>
<tr>
<td>Natural A/R at Pan</td>
<td>10000</td>
</tr>
<tr>
<td>From Past Facturing</td>
<td>85000</td>
</tr>
<tr>
<td>Total Available Cash</td>
<td>216600</td>
</tr>
<tr>
<td>QUARTERLY REPORT</td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td></td>
</tr>
<tr>
<td>TEAM NUMBER 2</td>
<td></td>
</tr>
<tr>
<td>GUEST NUMBER 10</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BEGINNING CASH ITEMS</th>
<th>SALES</th>
<th>M AND D</th>
</tr>
</thead>
<tbody>
<tr>
<td>M. IN HAND AT PAR</td>
<td>500</td>
<td>2</td>
</tr>
<tr>
<td>TOTAL AVAILABLE CASH</td>
<td>500</td>
<td>1</td>
</tr>
<tr>
<td>CURRENT FACTORING</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AT 90 PERCENT BLOCK 3</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>AT 90 PERCENT BLOCK 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL FACTORED ASH</td>
<td>500</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DISBURSEMENTS</th>
<th>TOTAL SALES</th>
</tr>
</thead>
<tbody>
<tr>
<td>FITS IN MC</td>
<td>144,100</td>
</tr>
<tr>
<td>VARIABLE COST</td>
<td>19,600</td>
</tr>
<tr>
<td>DIRECT SALARY</td>
<td>9,900</td>
</tr>
<tr>
<td>CONSTRUCTION COST</td>
<td></td>
</tr>
<tr>
<td>SALESMAN HIRE</td>
<td>1,000</td>
</tr>
<tr>
<td>M AND D EXPENSE</td>
<td>0</td>
</tr>
<tr>
<td>ADVERTISING EXPENSE</td>
<td>0</td>
</tr>
<tr>
<td>CONSULTING FEES</td>
<td>0</td>
</tr>
<tr>
<td>TOTAL DISBURSEMENTS</td>
<td>5,800</td>
</tr>
<tr>
<td>CASH LESS DISBURSEMENTS</td>
<td>12,400</td>
</tr>
<tr>
<td>PLANT VALUE</td>
<td>14,600</td>
</tr>
<tr>
<td>TOTAL SALESMAN</td>
<td>0</td>
</tr>
<tr>
<td>TOTAL AVAILABLE CASH</td>
<td>23,400</td>
</tr>
</tbody>
</table>

20 May 1963
APPENDIX III

MODIFICATIONS TO PROGRAM UMPIRE
The program modifications provided in this Appendix will serve to change the following limitations currently imposed by the program:

1) The number of teams playing.
2) The number of copies of the Quarterly Report produced.
3) The starting position of the teams.

The restriction to 10 customers sold per team per quarter is not easily modified, and will therefore not be included herein.

General

In general, the program modifications described here will require that the JOVIAL deck of the program be modified and re-compiled. The deck exists in two parts, each of which must be compiled separately. The last binary card produced by Part I must be removed and the two binary decks must be put together to produce the binary deck of UMPIRE for loading and operating. The modifications described here will all be done to the second part of the deck.

Number of Teams

In order to increase the maximum number of teams playing the game, the following changes must be made to the second part of the UMPIRE JOVIAL deck:

1) Replace the card reading as follows:
   TABLE QR V 7 S D $
   with a card reading as follows:
   TABLE QR V n S D $ where n = maximum number of teams playing.

2) Replace the card reading as follows:
   TABLE BOARD V 7 S D $
   with a card reading as follows:
   TABLE BOARD V n S D $ where n = maximum number of teams playing.

3) Alter preset cards as described below under "Starting Position."
Copies of Quarterly Report

In order to increase or decrease the number of copies of the Quarterly Report produced by program UMPIRE, the following change must be made:

In the Procedure defined by the following card:

```
PROC I'O'FOUR(INDEX) $
```

change the card currently reading as follows:

```
FOR J=0,1,2 $
```

to read as follows:

```
FOR J=0,1,n $ where n = the number of copies desired minus one.
```

This limit applies when no carbon paper is used on the printer. If two-part paper is used, twice as many copies will, of course, be produced.

Starting Position

Modifications to provide that the game begin at other points in time are more extensive than those described above, and require that the game actually be played up to the beginning point, with the board and Quarterly Report being maintained through this playing. It further requires, as does the changing of the number of teams, that the preset data for the items in the tables QR and BOARD be changed.

1) Preset Values

The preset values for the items listed below must be changed to reflect the situation at the beginning of the first quarter to be played. Other items need not be changed.

a. B'"CSH should reflect the Beginning Cash shown on the Quarterly Report for the first quarter to be played.

b. AR'PAR should reflect the Matured A/R @ $10,000 line under Starting Cash on the Quarterly Report for the first quarter to be played.

c. P'FACTR should reflect the Previous Factored A/R line under Starting Cash on the Quarterly Report for the first quarter to be played.

d. T'AV'CA should reflect the Total line under Starting Cash on the Quarterly Report for the first quarter to be played.

e. Alt' should reflect the number of units in the second block (Block 2) in the Accounts Receivable column on the game board at the beginning of the first quarter to be played.
f. AR'3 should reflect the number of units in the third block (Block 3) in the Accounts Receivable column on the game board at the beginning of the first quarter to be played.

g. AR'4 should reflect the number of units in Block 4 in the Accounts Receivable column on the game board at the beginning of the first quarter to be played.

h. AR'5 should reflect the number of units in Block 5 of the Accounts Receivable column on the game board at the beginning of the first quarter to be played.

i) INVTY should reflect the number of units in the Inventory block of the Production column on the game board at the beginning of the first quarter of play.

j) WIP should reflect the number of units in the Work in Process block of the Production column on the game board at the beginning of the first quarter of play.

k) OS'1 should reflect the number of units in the first block of the Construction column on the game board at the beginning of the first quarter of play.

l) CON'2 should reflect the number of units in the second block of the Construction column on the game board at the beginning of the first quarter of play.

m) CON'3 should reflect the number of units in the third block of the Construction column on the game board at the beginning of the first quarter of play.

n) FLD'1 should reflect the number of units in the field block (Block 1) of the Salesmen column on the game board at the beginning of the first quarter of play.

o) TRNG'2 should reflect the number of units in the second block of the Salesmen column on the game board at the beginning of the first quarter of play.

p) TRNG'3 should reflect the number of units in the third block of the Salesmen column on the game board at the beginning of the first quarter of play.
q) TRNG'4 should reflect the number of units in the fourth block of the Salesmen column on the game board at the beginning of the first quarter of play.

2) Preset Cards

The preset cards must conform to the following restrictions:

a) The cards must be punched only in columns 1 through 72.
b) The constants for preset must be punched as many times as the maximum number of teams playing.
c) The constants must be legal JOVIAL arithmetic constants with no fractional bits.
PART I
<table>
<thead>
<tr>
<th>Team</th>
<th>Team Number</th>
<th>Quarter Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Quarterly Report**

- **Starting Cash (SALES)**
- **Beginning Cash**
- **Accumulated U and D**
- **Product Improvement Achieved**
- **Total Available Cash**
- **Current Factoring**
- **At 90 Percent Block 2**
- **At 80 Percent Block 3**
- **At 70 Percent Block 4**
- **At 60 Percent Block 5**
<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>112P</td>
<td>Total Factorial A/R</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Total Sales</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Production</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Units in Inventory</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Sales Salaries</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Units in WIP</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Construction Cost</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Personnel</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Salesmen Lost</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Starting Capital</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>advertising expense</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Consulting fees</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>starting cash items</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Total Disbursements</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Remaining Cash</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>matured A/P at WIP</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Cash less Disbursements</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>In Hiring</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Prior Factoring</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Plant Value</td>
<td>$</td>
</tr>
<tr>
<td>112P</td>
<td>Total Available Cash</td>
<td>$</td>
</tr>
</tbody>
</table>
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM P5FACT R A 40 S S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM T5AVICA A 40 S S
BEGIN 4.0 4.0 4.0 4.0 4.0 4.0 4.0 END
ITEM BL319DP A 40 S S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM BL418DP A 40 S S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM BL518DP A 40 S S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM BL618DP A 40 S S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM TXICOFI A 40 S S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM IEL1COP A 40 5 S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM UX1199P A 40 5 S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM IL4NBOP A 40 6 S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM 169I8OP A 40 6 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM ANSFAC A 40 6 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM FXICOFI A 40 5 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM VICOSI A 40 5 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM TALAP A 40 6 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM CONSIA A 40 5 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM WIPE A 40 3 1
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM 40 A 40 5 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM CONSULT A 40 5 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM 016808 a 40 3 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM CAILSOB A 40 8 5
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM PAlES A 40 8 S
BEGIN 0.40 0.40 0.40 0.40 0.40 0.40 0.40 END
ITEM IM4Poy d 5
BEGIN 0

END TABLE CARD V 7 8 0 5
BEGIN
ITEM 012 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 013 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 014 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 015 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 016 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 017 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 018 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 019 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 020 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 021 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 022 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 023 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 024 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 025 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 026 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 027 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 028 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 029 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 030 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 031 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 032 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 033 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 034 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 035 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 036 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 037 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 038 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 039 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 040 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 041 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 042 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 043 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 044 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 045 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 046 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 047 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 048 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 049 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 050 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 051 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 052 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 053 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 054 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 055 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 056 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 057 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 058 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 059 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 060 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 061 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 062 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 063 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 064 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 065 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 066 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 067 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 068 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 069 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 070 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 071 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 072 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 073 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 074 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 075 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 076 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 077 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 078 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 079 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 080 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 081 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 082 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 083 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 084 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 085 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 086 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 087 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 088 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 089 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 090 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 091 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 092 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 093 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 094 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 095 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 096 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 097 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 098 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 099 A 10 U S
BEGIN 0.60 0.60 0.60 END
ITEM 100 A 10 U S
BEGIN 0.60 0.60 0.60 END
END TABLE A 10 U S
BEGIN
ITEM 101 A 9 U S
BEGIN 0.60 0.60 0.60 END
ITEM 102 A 7 U S
BEGIN 0.60 0.60 0.60 END
END BEGIN JUMPING
FOREST IF JUMPING B
BEGIN
CIRQUE = 0.6
JUMPING = WHICH BETS NEXT OR = NO OF TIMES
END FOR T = JB1MEN(B) - 1
BEGIN JIPROCESSING
JOSTIVITY B = JACCEPUS INPUTS
ACTIVITY B = JCOMPUTES FACTORED A/R
SPEEDY B = JCOMPUTES TOTAL DISBURSEMENTS
PROFITS B = JRESETS ACCELEBRATION R B AND E
Gрабатыва B = JDETERMINES SELLERS LOSS
END Y = JCOMPLETES OR SELLERS AND ADVANCE BARD
JOSTIVITY B = JRESETS OR FOR NEXT QUARTER
JOSTFIVE B = JOUTPUT OF ORS
END JIPROCESSING

BICEP FIRST S
ENC SUFFIRE
PNEP AC IREC (INDEX) S SICOMPUTES FACTORED AMM CORRECTS FOR OVERFACTORS
STEP INDEX A 49 S S
BEGIN
FOR I = INDEX S
BEGIN
IF BL 000 (SIS) GR 9000 OR AR J(SIS) S
BL J000 (SIS) = 9000 + AR J(SIS) S
IF BL J000 (SIS) = 9000 OR AR J(SIS) S
BL J000 (SIS) = 9000 + AR J(SIS) S
IF BL J000 (SIS) = 9000 OR AR J(SIS) S
BL J000 (SIS) = 9000 + AR J(SIS) S
IF BL J000 (SIS) = 9000 OR AR J(SIS) S
BL J000 (SIS) = 9000 + AR J(SIS) S
AR JFACT (SIS) = BL J000 (SIS) + BL J000 (SIS) + BL J000 (SIS)
END
END
ENC PNEP SPEND (INDEX) S
STEP INDEX A 49 S S
BEGIN
FOR I = INDEX S
BEGIN
FX ICOST (SIS) = CUST FIXED (SUBJ (SIS)) S
VICOST (SIS) = VIP (SIS) + COSTVAR (SUBJ (SIS)) S
SALARY (SIS) = 1000 + FLD (A) = TRNOB (SIS) + TRNOJ (SIS)
+ TRNOK (SIS) S
AD (SIS) = 3000 + PACT (SIS) S
CIS (SIS) = FXICOST (SIS) + MICOST (SIS) + SALARY (SIS)
+ CONS (SIS) + MIRE (SIS) + AD (SIS) + CONSULT (SIS) S
END
END
ENC PNEP PAVE (INDEX) S SNEWATES ACCUMULATED R AND D S S
STEP INDEX A 49 S S
BEGIN
FOR I = INDEX S
BEGIN
IF IMP (SIS) OR AD (SIS) EQ 0 S
BEGIN
CUMRD (SIS) = 0 S
RETURN S
END
CUMRD (SIS) = CUMRD (SIS) + AD (SIS) S
END
END
ENC PNEP QUITTER (INDEX) S SICATROLS SALESMAN LOSSES S S
STEP INDEX A 49 S S
BEGIN
FOR I = INDEX S
BEGIN
MIRE (SIS) = MIRE (SIS) / 1000 S
IF FLD (A) = 0 S
BEGIN
FLD (A) = FLD (A) + FLD (A) S
END
END

FOR I = INDEX B
  BEGIN
    IF I EQ 0 THEN
    BEGIN
      LOG(10)(QUARTER NUMBER) = 6
      FLEXIN B
      BYTE (572,2560) = RESPOND 8
      TEMP = DTI(RESPOND) = 1
      BYTE(572,2560) = STD(TEMP) S
    END
    BYTE(572,TEAMNO) = STD(I = 1) S
    LOG(TEAMNO) = 5
    LOG(UNITS SCHEDULED) = 5
    FLEXIN B
    SCHED(SIS) = DTI(RESPOND) S
    LOG (1) = BLOCK 2 FACTORING) = 5
    FLEXIN B
    BLO9B(SIS) = DTI(RESPOND) S
    LOG (7) (BLOCK 3) = 5
    FLEXIN B
    BLO9B(SIS) = DTI(RESPOND) S
    LOG (7) (BLOCK 4) = 5
    FLEXIN B
    BLO9B(SIS) = DTI(RESPOND) S
    LOG (7) (BLOCK 5) = 5
    FLEXIN B
    BLO9B(SIS) = DTI(RESPOND) S
    LOG (7) (BLOCK 6) = 5
    FLEXIN B
    BLO9B(SIS) = DTI(RESPOND) S
    LOG (1) (CONSTRUCTION) = 5
    FLEXIN B
    CON()B(SIS) = DTI(RESPOND) S
    LOG (13) (CONSULTING) = 5
    FLEXIN B
    CONSULT(SIS) = DTI(RESPOND) S
    LOG (13) (ADVISOR) = 5
    FLEXIN B
    PAGIE(SIS) = DTI(RESPOND) S
    LOG (1) (SALES) = 5
    FLEXIN B
    SALES(SIS) = DTI(RESPOND) S
    IF SALES(SIS) NO 0 THEN
    BEGIN
      LOG (8) (CUSTOMERS AND UNITS SOLD) = 0
      FOR J = 0 TO 9
      BEGIN
        FLEXIN B
        CUST(J) = DTI(RESPOND) S
        IF CUST(J) EQ 0 THEN
          GOTO SETO S
        FLEXIN B
        SOLD(J) = DTI(RESPOND) S
      END
TEST 5
BEGIN
  FOR K = 1 TO 10
      CUST(K) = 0
      SOLD(K) = 0
  END FOR
END

NEXT
LOG(7-PROMOTIONS) 2 OR 0
FLEX
CUST(7) = 0
SOLD(7) = 0
END

NEXT
LOG(7-PROMOTIONS JUNKED) 2
FLEX
CUST(7) = 0
SOLD(7) = 0
END

END

IF CUST(1) = 0
BEGIN
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(3) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(4) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(5) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(6) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(7) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
  END
ENDIF

IF CUST(1) = 0
BEGIN
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(3) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(4) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(5) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(6) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
  END
ENDIF

IF CUST(1) = 0
BEGIN
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(3) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(4) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(5) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(6) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
  END
ENDIF

IF CUST(1) = 0
BEGIN
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(3) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(4) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(5) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(6) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
  END
ENDIF

IF CUST(1) = 0
BEGIN
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(2) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(3) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(4) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(5) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
    GOTO 810
  END
  IF CUST(6) = 0
  BEGIN
    BYTE(0,20)(GR14) = STD(CUST(20))
    BYTE(0,20)(GR15) = STD(SOLD(20))
  END
ENDIF
```
BYTE(3152.85119354) = STD(AR#FXR(825))  
BYTE(3156.8511954) = STD(AR#FXR(825))  
BYTE(3152.8511956) = STD(AR#FXR(825))  
BYTE(3152.8511958) = STD(AR#FXR(825))  
OPEN OUTPUT ORPRINT 5
 FOR J = 0 TO 2 5
 BEGIN
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 OUTPUT ORPRINT BLANK 5
 ```
20 May 1963

```
TIMEDIV(1) & (OUTPUT OF DRI)

FUNCTION

BITP FIRST
ENCC JUMPINDEX
PENC ACTIVEX(INDEX) & (COMPUTES FACTORED A/R CORRECTS FOR AVERFACTORS)
BEGIN
FOR I = INDEX TO
BEGIN
BEGIN IF AL2190P(INX) = 9000 * AR13(INX)
AL2190P(INX) = 9000 * AR13(INX)
IF AL3190P(INX) = 9000 * AR13(INX)
AL3190P(INX) = 9000 * AR13(INX)
IF AL4190P(INX) = 9000 * AR13(INX)
AL4190P(INX) = 9000 * AR13(INX)
IF AL5190P(INX) = 9000 * AR13(INX)
AL5190P(INX) = 9000 * AR13(INX)
IF AL6190P(INX) = 9000 * AR13(INX)
AL6190P(INX) = 9000 * AR13(INX)
IF AL7190P(INX) = 9000 * AR13(INX)
AL7190P(INX) = 9000 * AR13(INX)
END
END
PENC INDEX A = 4 + 5
BEGIN
FOR I = INDEX TO
BEGIN
BEGIN IF FL12190P(INX) = 9000 * AR13(INX)
FL12190P(INX) = 9000 * AR13(INX)
IF FL2190P(INX) = 9000 * AR13(INX)
FL2190P(INX) = 9000 * AR13(INX)
IF FL3190P(INX) = 9000 * AR13(INX)
FL3190P(INX) = 9000 * AR13(INX)
IF FL4190P(INX) = 9000 * AR13(INX)
FL4190P(INX) = 9000 * AR13(INX)
IF FL5190P(INX) = 9000 * AR13(INX)
FL5190P(INX) = 9000 * AR13(INX)
IF FL6190P(INX) = 9000 * AR13(INX)
FL6190P(INX) = 9000 * AR13(INX)
IF FL7190P(INX) = 9000 * AR13(INX)
FL7190P(INX) = 9000 * AR13(INX)
END
END
PENC INDEX A = 4 + 5
BEGIN
FOR I = INDEX TO
BEGIN
BEGIN IF PM12190P(INX) = 9000 * AR13(INX)
PM12190P(INX) = 9000 * AR13(INX)
IF PM2190P(INX) = 9000 * AR13(INX)
PM2190P(INX) = 9000 * AR13(INX)
IF PM3190P(INX) = 9000 * AR13(INX)
PM3190P(INX) = 9000 * AR13(INX)
IF PM4190P(INX) = 9000 * AR13(INX)
PM4190P(INX) = 9000 * AR13(INX)
IF PM5190P(INX) = 9000 * AR13(INX)
PM5190P(INX) = 9000 * AR13(INX)
IF PM6190P(INX) = 9000 * AR13(INX)
PM6190P(INX) = 9000 * AR13(INX)
IF PM7190P(INX) = 9000 * AR13(INX)
PM7190P(INX) = 9000 * AR13(INX)
END
END
```

TM-1088/002/00
FIGURE(F114(S15) = FLD114(S15),FLD14(S15)) S
IF TANG12(S15) = 0 0 0 S
FIGUR(F114(S15) = TANG12(S15),FLD14(S15)) S
IF TANG13(S15) = 0 0 0 S
FIGUR(F114(S15) = TANG13(S15),FLD13(S15)) S
IF TANG14(S15) = 0 0 0 S
FIGUR(F114(S15) = TANG14(S15),FLD13(S15)) S
IF TANG15(S15) = 0 0 0 S
FIGUR(F114(S15) = TANG15(S15),FLD13(S15)) S

EAU
END
PNC FIGURE(EU41N = NUM41UT,LOSSES) S JUD CDE TURNS SALESMAN LOSSES
IPE 1 U 7 U 5 S
IPE 1 0 8 S
IPE 1 0 8 S
IPE 1 0 8 S
IPE 1 0 8 S
IPE 1 0 8 S
IPE 1 0 8 S
BEGIN
LOSSES = 0 0
NUM41UT = NUM41UT
FOR I = NUM41UT - 1, 1, 0 S
BEGIN
RANGEN(RANDO) S
IF RANDO < 0 S
BEGIN
NUM41UT = NUM41UT - 1 S
LOSSES = LOSSES + 1 S
END
END
PNC RANGEN(RANDO) S IF USING RANDOM NUMBER GENERATOR
TABLE GEN A 7
BEGIN
IPE GEN A 4 0 0 8
DEBUG 01025434343010467,01272343473274254 END
IPE 1 TEMPA S 7 U 7 I OUTPUT PARAMETERS
IPE 1 TEMPA S 6 4 S
BEGIN
LET, TEMPA = GEN18(SUB) + GEN41UT
GEN18(SUB) = GEN10,27(SUB)(TEMPA) S
PNC PUN = TIT(SUB),SUB1GEN(SUB) S
IF PUN = 99,40 S
GETS GEN41UT S
END
PNC GETS
IPE UTC = 8
IPE 1 UA A 4A 55
IPE 1 UX A 4A 55
BEGIN PNC PRINTS ALPHABETIC TO DECIMAL
DIRECT
ARCG 0,1 S
TDE RTR,ER S
TDE 0A000000000000000000 S
TDE WTR,WD S
TDE 0,1 S
CITA 1TE,NN S
END
PNC 1TE,NN S
END
PNC WTR,WD S
END
PNC RTR,ER S
END
20 May 1963

```
A1025 B2031 C
A1025 (P)+3H,  S
A1077 C/MT:61/CHLT,(P)-6HS
A1077 6,1,  S
A1077 MT:492,  S
A1077 0,1,  S

JVIAL
ENC 1STDIEN
PNC UTG:RAS
ITEM UTG A 47 JS
ITEM MM = AR
ITEM MUNITEN A 47 JS
BEGIN ADJUST MM HM TO ETHEY BSTD
DIRECT
C4 UT-END  SOLAR HTA.
T9  UT1842 SET L INTO MUNITEN FOR
T9  UT:PLANLIST MULTIPLICATION.
C6  UT:0 SOLAR U.
T2  UT:HM SET HM CYCLE HIGH
S2  D  BAND PLACE BACK IN HM,
T2  D  HM "
T2  D  SHIFT ACTIVE BYTE
T4  0  INT  SINTO THE SCALED HM IN 0
J240 (P)+3H SKIP BLANK SKIP BYTE.
H1 UT:PLANMULT:HT MUNITEN SCALED
A6  UT:PLANB:PRODUCT MUNITEN IN A.
T-1 C/MT:61/CHLT MUNITEN BY 50 AND
H4  UT:PLANM1E NITY HM1N
F1  D/ER MUNITEN100,000,000
J2BO (P)+4H SHARING LAST BYTE,
S2  D  HSCALE MUNITEN TO HM2
A4 UT:PLAN
J20  (P)+5H REPEAT FOR NEXT HM.

JVIAL
ENC 1STDIEN
PNC UTG:RAS
ITEM UTG:RB
ITEM INDEX A 41 9 4
BEGIN
FOR I = INDEX 6
BEGIN
IF SALES(IS1S) OR INVY(IS1S) 8
BEGIN
SALES(IS1S) = INVY(IS1S) 8
COSTPERSALES HAVE BEEN REDUCED TO THE NUMBER OF UNITS IN INVY
TOP1) IS
END
INVY(IS1S) = INVY(IS1S) - SALES(IS1S) 8
CAH1(IS1S) = AR1(IS1S) 8
AR1(IS1S) = AR1(IS1S) 8
AR2(IS1S) = AR2(IS1S) 8
AR1(IS1S) = AR1(IS1S) 8
AR3(IS1S) = SALES(IS1S) 8
RENT(IS1S) = FLUPS(IS1S) + TANQ1(IS1S) + TANQ1(IS1S) + TANQ1(IS1S)
+ TANQ1(IS1S) 8
CALDOR1(IS1S) = TAH1(IS1S) + BIRD1(IS1S) 8
INVY(IS1S) = INVY(IS1S) + WIP1(IS1S) 8
WIP1(IS1S) = INCOME1(IS1S) 8
EXEC1(IS1S) 8
COSTS1S) 8
COSTS1S) 8
CO1(IS1S) 8
CO2(IS1S) 8
```
20 May 1963

BEGIN
IF I EQ "S"
BEGIN
LOGVAL (PLEASE ENTER THE FOLLOWING VALUES): $.
LOG (54) = (JSTHEN LPMEM) $.
FLXIN $.
BYTE (172) = GRET $.
TEMP = DT (RENDEG) $.
BYTE (49) = GRQ (LH$).

END
BYTE (55) (TERM US) = DT (I + 1) $.
LOG (TERM US) $.
LOG (47) (UNITS SCHELLEN) $.
FLXIN $.
SCH (1) = DT (SHUSPND) $.
LOG (1) $.
LOG (12) (LOCK 2, FACT:USPND) $.

FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (LOCK 3) $.
FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (LOCK 4) $.

FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (LOCK 5) $.
FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (LOCK 6) $.

FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (MULTI G) $.
FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (MULTI H) $.

FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (MULTI I) $.
FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (MULTI J) $.

FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (MULTI K) $.
FLXIN $.
813050 (5) = DT (SHUSPND) $.
LOG (7) (MULTI L) $.

BEGIN
LOG (51) (DEVS (AND WORKS HUND)) $.
FCH = 0.004 $.
GOTO ; S.

FLXIN $.
CLS (5) = DT (SHUSPND) $.
IF CNT (LH$) = 0 $.
GOTO ; SET $.
FLXIN $.
SCH (5) = DT (SHUSPND) $.
20 May 1963

```
TEST

GETA FOR K = .1/.6 $.
RETL
CUST(SK1) = 0 $.
SHLNT(SK1) = 0 $.
FND
GETC NEXT $.
END

ALN1. LIGHT (PRODUCT IMPRESSION, 1 OR 1)) $.
LEN =
TYPH(SIS) = JTR(NRSCHN)) $.
LFT(NR) LINES LACED) $.
FIND $.
TEMP = CTR(46SPRINT) $.
PLVAL(SIS) = PLVAL(SIS) = 3000 + TEMP $.
CSF(SIS) = TSL(SIS) = TEMP $.
FND $.
END

PHEC. LIGHT (PRODUCT IMPRESSION) $.
INJ - INDEX A = 40 $.
REN $.
F(A) = INDEX $.
-EXIT $.
IF CUST(40) EQ 2 $.
GOTO 414 $.
BYTE$(40, 23)(OR14) = BTDICUST(S461) $.
BYTE$(40, 23)(OR14) = BTDCUST(S46) $.
IF CUST(41) EQ 0 $.
GOTO 414 $.
BYTE$(40, 23)(OR14) = TDICUST(S145) $.
BYTE$(40, 23)(OR16) = BTDCUST(S154) $.
IF CUST(24) EQ 0 $.
GOTO 514 $.
BYTE$(40, 23)(OR14) = BTDCUST(S154) $.
BYTE$(40, 23)(OR14) = BTDCUST(S145) $.
IF CUST(43) EQ 0 $.
GOTO 534 $.
BYTE$(40, 23)(OR21) = BTDCUST(S154) $.
BYTE$(40, 23)(OR21) = BTDCUST(S35) $.
IF CUST(44) EQ 0 $.
GOTO 534 $.
BYTE$(40, 23)(OR22) = BTDCUST(S154) $.
BYTE$(40, 23)(OR22) = BTDCUST(S35) $.
IF CUST(45) EQ 0 $.
GOTO 534 $.
BYTE$(40, 23)(OR23) = BTDCUST(S35) $.
BYTE$(40, 23)(OR23) = BTDCUST(S154) $.
IF CUST(46) EQ 0 $.
GOTO 534 $.
BYTE$(40, 23)(OR24) = BTDCUST(S35) $.
BYTE$(40, 23)(OR24) = BTDCUST(S154) $.
```

20 May 1963

```
OUTPUT QPRINT 2804 
OUTPUT QPRINT 2803 
OUTPUT QPRINT 2802 
OUTPUT QPRINT 2801 
OUTPUT QPRINT BLANK 8 
OUTPUT QPRINT 2809 
PRESQPRINT = 0 3
END
SHUT OUTPUT QPRINT 8
END

PROC LOG(FLXING)5
TEPP FLXING = 48ST US 1FLEX IC IMAGE
ITC AR A A 48 ST SAVE INDEX REGS 1 AND 2 HERE11
BEGIN 1FRON T TO PROG CUT UP TO 120 CHARACTERS ON THE FLEXO SUP-
PRESSING LEADING BLANKS11

DIRECT:

L001 TTK 1.1
L002 TTK 2.2
L003 TTA 3.3
L004 TNA 4.4

FINAL

LFRlIQUE1
PXG. FLEIXING
TEEP = A 48 S $ 3 SAVE INDEX REGS 1 HERE11
ITEP = A 48 S $ 3 TEMPORARY STORAGE...DIRECTO I OF TCH INSTRUCTIONS
BEGIN 1FRON T TO INPUT UP TO 120 CHARACTERS FROM FLEXO. STOP CODE
ALLOWS RESTANT FOR ERRORS. CARRIAGE RETURN ENDS PASSAGE11

EXITQ
TCLQ 0 1
T 1 F 6
AA C17 UPDIRE, REBON
```
System Development Corporation,
Santa Monica, California

BUSINESS MANAGEMENT GAME, PART III:
INSTRUCTIONS FOR THE USE AND MODIFICATION OF PROGRAM UMPIRE. Scientific
rpt., TM-1088/002/00, by S. Peterson.
20 May 1963, 64p.

Unclassified report

DESCRIPTORS: Management Engineering.

Describes the use, modification and
maintenance of program UMPIRE, a pro-
gram written in JUVIAL for the Philco
2000, to be used in the play of the
management game described in TM-1088.