Development and Testing of a Heuristic Line Balancing Program for a Microcomputer

1986

Dean B. Creech
University of Central Florida
DEVELOPMENT AND TESTING OF A HEURISTIC LINE BALANCING PROGRAM FOR A MICROCOMPUTER

BY

DEAN BRADLEY CREECH
B.S., University of Texas, 1979

RESEARCH REPORT

Submitted in partial fulfillment of the requirements for the degree of Master of Science in the Graduate Studies Program of the College of Engineering University of Central Florida Orlando, Florida

Fall Term
1986
Development, operation, and testing of a heuristic line balancing program for a microcomputer are discussed. Tasks are grouped into work stations along an assembly line such that the number of work stations required is minimized.

The model is built primarily using the Hoffmann (1963) procedure with modifications described by Gehrlein and Patterson (1975). For purposes of comparison the Rank Positional Weight technique (Helgeson and Birnie, 1961) is also included in the model.

Testing included thirty-seven different balances using problems from the literature. For each balance both Rank Positional Weight and Hoffmann solutions were obtained in the forward and reverse directions.

Four measures of performance were considered in this study. These measures of performance were: (1) the average percentage a balance is above the optimum solution, in terms of number of stations; (2) time to obtain a balance; (3) the best solution in terms of the lowest number of stations and lowest standard deviation of the slack times; and (4) the largest value of minimum station slack time. Overall it was found that the
Hoffmann procedure with a delay factor of 1.5 was best suited for the microcomputer application. Further work is recommended to find the optimum delay factor and apply the Modified Hoffmann procedure to solving line balancing problems where the cycle time is minimized given a set number of work stations.
# TABLE OF CONTENTS

LIST OF TABLES ........................................... v
LIST OF FIGURES .......................................... vi

Chapter

I. INTRODUCTION ............................................. 1
II. HEURISTIC LINE BALANCING TECHNIQUES
    INVESTIGATED ........................................... 3
III. PROGRAM DEVELOPMENT AND USE ......................... 16
IV. TESTING AND RESULTS ................................ 30
V. CONCLUSIONS AND RECOMMENDATIONS FOR
    FURTHER WORK ........................................... 37

Appendix

BALANCE PROGRAM LISTING ................................ 39
LIST OF REFERENCES ....................................... 73
# LIST OF TABLES

1. Rank Position Weight Balance.  
   8
2. Hoffmann Balance for Problem in Figure One.  
   12
3. Modified Hoffmann Balance for Problem in  
   Figure One.  
   14
4. Comparison of the Average Percent Above Optimum  
   Performance in Terms of Number of Stations.  
   34
5. Number of Best Solutions Out of 37 Balances When  
   Both Number of Stations and Standard Deviation  
   of Slack Times are Considered  
   34
LIST OF FIGURES

1. Precedence Diagram for an Assembly Line .......... 6
2. Main Program Menu .................................. 21
3. Sample Program Output ............................... 26
4. Auto Test Generator Menu ......................... 26
I. INTRODUCTION

When developing a production line one must determine how to assign tasks to work stations such that: (1) the time allotted at any one station does not exceed a predetermined amount, called the cycle time; (2) no task is split between two work stations; and (3) precedence relationships among work elements are not violated. In addition, it may be required that certain tasks be performed (or not performed) at the same work station (zoning constraints), or that the balance delay present in a specified balance is divided equally among all work stations.

There are two basic types of assembly line balancing problems discussed in the literature. The solution of a Type I problem is the assignment of tasks to work stations that minimizes the number of stations required for a given production rate (cycle time). The Type II problem solution is the assignment of tasks to work stations that maximizes the production rate (minimizes cycle time) for a given number of work stations. Several methods for solving the line balancing problem have been developed. Most of these procedures solve the Type I problem. Type II problems are usually solved by iteratively solving the
Type I problems and increasing the cycle time after each iteration.

This paper reports on the development and testing of a line balancing program for a personal microcomputer that solves the Type I problem. The objective was to develop a program that would fit in the standard memory map of an Apple IIc computer. Test results are compared with results previously reported in the literature. In the conclusion, results are summarized and recommendations are made for further work.
II. HEURISTIC LINE BALANCING TECHNIQUES INVESTIGATED

In this section the heuristic decision rules investigated are described. Additionally, the reasons for selecting these methods are also discussed.

Justification of Heuristic Methods Selected

A comparative evaluation of twenty-four line balancing techniques that solve Type I problems was conducted by Talbot, Patterson, and Gehrlein (1986). This evaluation included 1604 different balances. The evaluation found that the Hoffmann technique (1963) with modifications described by Gehrlein and Patterson (1975) would be very well suited for a personal computer based on memory storage requirements and time to perform the balance.

Talbot, Patterson, and Gehrlein reported that the Modified Hoffmann technique, with a modifier of one, performed very well in terms of CPU time requirements and balances obtained. On the average the Modified Hoffmann technique found solutions that were only 0.36% above optimum. In terms of CPU time, the Modified Hoffmann
procedure found balances in 32% of the time required by routines that find optimal solutions.

In order to compare the performance of the Hoffmann and Modified Hoffmann procedure to another method, the Rank Positional Weight technique (Helgeson and Birnie, 1961) was also included in the program and tested. Results of these methods are compared to check for concurrence with Talbot, Patterson, and Gehrlein.

**Description of the Rank Position Weight Technique**

Before tasks are assigned to a station using the Rank Position Weight (RPW) technique, the precedence relationships and positional weight for an activity are determined.

**Calculating Precedence Numbers**

The precedence relationships establish the order in which tasks must be performed as a product moves down the assembly line. For programing considerations, the standard method has been to use a precedence matrix (Mansoor 1964). If activity "i" immediately precedes activity "j" then the precedence matrix value for column "i" row "j" is one, otherwise its value is zero. The columns are then summed to obtain the initial precedence numbers. Those columns whose sum is zero represent activities whose precedence restrictions have been
satisfied. When a task is selected and assigned to a station, then the precedence numbers of those activities that immediately follow the activity selected are decreased by one.

Calculating Positional Weights

Once the precedence relationships have been established then the positional weights for each of the activities are determined. The positional weight for any work unit is obtained by adding its own time to that of all the dependent operations performed after it. Applying this to the assembly line shown in Figure 1 (Mansoor 1964), then the positional weights are as follows (Positional Weight = PW and Activity = ACT):

- PW of ACT 6 = 1
- PW of ACT 5 = 10 + PW of ACT 6 = 11
- PW of ACT 4 = 2 + PW of ACT 5 = 13
- PW of ACT 3 = 7 + PW of ACT 4 = 20
- PW of ACT 2 = 12 + PW of ACT 3 = 32
- PW of ACT 1 = 8 + PW of ACT 6 = 9

Activity Assignment Rules

After the precedence and positional weights have been determined, activities are assigned to work stations. The rules outlined by Helgeson and Birnie (1961), with modifications required for microcomputer use, are reproduced here.
Figure 1. Precedence Diagram for an Assembly Line, Showing the Precedence Relationships Between Activities (Numbers in Circles) and Their Corresponding Times (Numbers Outside Circles)
1. Select the work unit with the highest positional weight whose precedence number is zero to the first work station.

2. Determine new precedence numbers based on the task assigned. Reduce the precedence numbers of immediately following tasks by one. Set the precedence number of the task assigned to negative ten.

3. Calculate the unassigned time for the work station by subtracting the task time of the activity from the station time remaining before the assignment.

4. Select the work unit with the next highest positional weight and attempt to assign it to the work station after making the following checks:
   a. Check to ensure the activities precedence number is zero.
   b. Compare the work unit time with the unassigned time for that station. If the task time is less than the time remaining then select the work unit for assignment, recalculate time remaining, and then go to step five. If the work unit time is greater than the unassigned time, proceed to step five.

5. Continue to select, check and assign work units until no unassigned task remains that can satisfy both the precedence and time requirements.

6. Repeat steps one through five for the next work station.

7. Continue assigning work units to work stations until all activities have been assigned.

Sample Solution

The problem is to balance the assembly line illustrated in Figure 1. The problem is analyzed for a cycle time equal to fifteen. Table 1 summarizes the results of this balance using the RPW solution method.
<table>
<thead>
<tr>
<th>STATION NUMBER</th>
<th>TIME LEFT BEFORE SELECTION</th>
<th>TASK SELECTED</th>
<th>TIME LEFT</th>
<th>TASKS ASSIGNED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td>3</td>
<td>8</td>
<td>3,4</td>
</tr>
<tr>
<td>3</td>
<td>14</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>14</td>
<td>1</td>
<td>7</td>
<td>1,6</td>
</tr>
</tbody>
</table>

NO OTHER TASK SATISFIES PRECEDENCE AND TIME REQUIREMENTS. GO TO NEXT STATION.

BALANCE COMPLETE
Description of the Hoffmann Technique

Like the RPW technique, the Hoffmann method of assembly line balancing starts by determining precedence numbers. However, since the Hoffmann technique uses backtracking to find the best possible balance for a station the manipulation of the precedence numbers during the procedure is handled differently. Three sets of precedence numbers are maintained. The initial precedence numbers are those determined as described above in the RPW solution. A current set of precedence numbers is maintained that corresponds to the best selection of activities for a station. During the selection of the best assignment of work units to a station, a test precedence number is maintained. The test precedence number corresponds with the current test solution.

Activity Assignment Rules

After establishing the initial precedence numbers, activities are assigned to work stations. The scheme for generating the feasible combinations and balancing the line station by station, as discussed by Hoffmann (1963) and modified for microcomputer use, is reproduced here.

1. Search the precedence numbers for a zero.
2. Select the work element whose precedence number is zero.
3. If the selected element's time is less than or equal to the time remaining then go to step four. Otherwise go to step six.
4. Establish the new test precedence numbers by subtracting one from the precedence number of those activities whose immediate preceding work element was the one just selected. Set the precedence number of the element just selected to negative ten.

5. Subtract the task time of the element just selected from the cycle time remaining.

6. Go back to step one and continue searching for activities to be assigned to work stations. For each precedence number the search starts with the precedence number for activity one. Each time the search returns to step one the search begins one activity number higher than it was in the previous search of this precedence number. When all columns have been examined and no other elements can be selected then go to step seven.

7. Compare the slack time of the test solution established in steps one through six to the slack time of the current solution. If the slack time of the test solution is less than the slack time of the current solution then the test solution becomes the current solution. Otherwise the current solution is maintained.

8. The search now backtracks one precedence number and returns to step one. To backtrack the precedence number of the element just selected is set at zero and one is added to the precedence number of those activities that immediately followed the activity just selected. The search at step one is then begun at one element to the right of the element that was last selected.

9. Steps one through eight are repeated until the search has enumerated all possible task assignments for a work station. This enumeration is complete when the search of the first precedence number for the station is at the last activity.

10. The station number is incremented by one and steps one through nine are repeated for the next station.

11. Steps one through ten are repeated until all activities have been assigned to a work station. This occurs when all precedence numbers are equal to negative ten.
Sample Solution

The procedure described above is applied to the precedence diagram given in Figure 1. A cycle time of fifteen is used. The results are summarized in Table 2.

Description of a Modification to the Hoffmann Procedure

In the solution of a Type I problem the quantity

\[ BD = (C)(N) - ST \]

where BD = Balance Delay  
C = Cycle Time  
N = Number of Stations  
ST = Sum of the Individual Task Times

represents the balance delay which is necessarily present in a balance. A disadvantage of the Hoffmann procedure discussed in the previous section is that the earlier stations in the assignment have a tendency to be more heavily loaded in the respect that the latter stations are more likely to have the greater amount of balance delay. It is desirable to have the slack time at a station allocated evenly among all work stations. The Hoffmann procedure can be modified to terminate the search for an assignment of tasks to a work station whenever a feasible assignment of tasks to that station results in a slack
<table>
<thead>
<tr>
<th>K</th>
<th>PRECEDENCE NUMBERS</th>
<th>TASK SELECTED</th>
<th>TASK TIME</th>
<th>STATION TIME BEFORE</th>
<th>STATION TIME AFTER</th>
<th>TASKS ASSIGN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0 0 1 1 1</td>
<td>1 8 15</td>
<td>7</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-D 0 1 1</td>
<td>2 12 7</td>
<td>-</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CURRENT SOLUTION</td>
<td>7</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0 0 1 1</td>
<td>2 12 15</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-D 0 1</td>
<td>1 8 3</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-D 0 1</td>
<td>3 7 3</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CURRENT SOLUTION CHANGED SINCE 3&lt;7</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0 0 1 1</td>
<td>15</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATION 1 SELECTION COMPLETE</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-D 0 1</td>
<td>1 8 15</td>
<td>7</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-D -D 0 1</td>
<td>3 7 7</td>
<td>0</td>
<td>1,3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-D -D -D 0</td>
<td>4 2 0</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CURRENT SOLUTION</td>
<td>0 1,3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-D -D 0 1</td>
<td>4 2 8</td>
<td>6</td>
<td>3,4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-D 0 1</td>
<td>3 7 8</td>
<td>3</td>
<td>1,3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-D -D 0 1</td>
<td>3 7 8</td>
<td>0</td>
<td>1,3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-D -D -D 0</td>
<td>4 2 0</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAME AS CURRENT SOLUTION</td>
<td>0 1,3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-D -D 0 1</td>
<td>4 2 8</td>
<td>6</td>
<td>3,4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0 -D -D -D 0</td>
<td>1 8 6</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0 -D -D -D 0</td>
<td>5 10 6</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TEST SOLUTION REJECTED 6&gt;0</td>
<td>2 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-D -D 0 1</td>
<td>8 3</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATION TWO COMPLETE</td>
<td>15</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-D -D -D 0 1</td>
<td>4 2 15</td>
<td>13</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-D -D -D 0 1</td>
<td>5 10 13</td>
<td>3</td>
<td>4,5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-D -D -D -D 0</td>
<td>6 1 3</td>
<td>2</td>
<td>4,5,6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>-D -D -D -D -D 0</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALL PRECEDENCE NUMBERS EQUAL NEGATIVE TEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BALANCE COMPLETE.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATION THREE COMPLETE</td>
<td>2</td>
<td>4,5,6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
time that satisfies the following relation (Gehrlein and Patterson 1975):

\[ TL \leq DF(BD/N) \]

where \( TL \) = Time left at a station (slack time),
\( DF \) = Delay Factor
\( BD \) = Balance Delay
\( N \) = Number of Work Stations

Effectively this establishes what percentage of the average balance delay is acceptable at a station. The delay factor is then varied in an attempt to obtain alternate balances with less variation in slack time between stations. Given several balances yielding the same number of work stations, the balance with the minimum standard deviation of slack times can be selected.

Sample Solution

The Modified Hoffmann procedure is applied to the precedence diagram given in Figure 1. A delay factor of 1.5 is used and the balance is solved with a cycle time of fifteen. The acceptable delay at a station is then given by:

\[ DF(BD/N) = 1.5(5/3) = 2.5 \]

The results of the balance are summarized in Table 3. By comparing the balances given in tables 2 and 3, another key advantage of the Modified Hoffmann procedure
# Table 3

Modified Hoffmann Balance for Problem in Figure One

**Delay Factor** = 1.5  
**Cycle Time** = 15

<table>
<thead>
<tr>
<th>K</th>
<th>Precedence Numbers</th>
<th>Task Selected</th>
<th>Task Time</th>
<th>Station Time Before</th>
<th>Station Time After</th>
<th>Tasks Assign</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0 0 1 1 1 2</td>
<td>1</td>
<td>8</td>
<td>15</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>-D 0 1 1 1 1</td>
<td>2</td>
<td>12</td>
<td>7</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**Current Solution**

<table>
<thead>
<tr>
<th>K</th>
<th>Precedence Numbers</th>
<th>Task Selected</th>
<th>Task Time</th>
<th>Station Time Before</th>
<th>Station Time After</th>
<th>Tasks Assign</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0 0 1 1 1 2</td>
<td>2</td>
<td>12</td>
<td>15</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>0 -D 0 1 1 2</td>
<td>1</td>
<td>8</td>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Current Solution Changed Since 3 < 7

<table>
<thead>
<tr>
<th>K</th>
<th>Precedence Numbers</th>
<th>Task Selected</th>
<th>Task Time</th>
<th>Station Time Before</th>
<th>Station Time After</th>
<th>Tasks Assign</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0 0 1 1 1 2</td>
<td>3</td>
<td>7</td>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Station 1 Selection Complete

<table>
<thead>
<tr>
<th>K</th>
<th>Precedence Numbers</th>
<th>Task Selected</th>
<th>Task Time</th>
<th>Station Time Before</th>
<th>Station Time After</th>
<th>Tasks Assign</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0 -D 0 1 1 2</td>
<td>1</td>
<td>8</td>
<td>15</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>-D -D 0 1 1 1</td>
<td>3</td>
<td>7</td>
<td>7</td>
<td>0</td>
<td>1,3</td>
</tr>
</tbody>
</table>

Slack Time ≤ Acceptable Delay

<table>
<thead>
<tr>
<th>K</th>
<th>Precedence Numbers</th>
<th>Task Selected</th>
<th>Task Time</th>
<th>Station Time Before</th>
<th>Station Time After</th>
<th>Tasks Assign</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-D -D -D 0 1 1</td>
<td>4</td>
<td>2</td>
<td>15</td>
<td>13</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>-D -D -D -D -D 0 1</td>
<td>5</td>
<td>10</td>
<td>13</td>
<td>3</td>
<td>4,5</td>
</tr>
<tr>
<td>3</td>
<td>-D -D -D -D -D -D 0</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4,5,6</td>
</tr>
</tbody>
</table>

All Precedence Numbers Equal Negative Ten

Balance Complete.

Station Three Complete

<table>
<thead>
<tr>
<th>Tasks Assign</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
</tr>
</tbody>
</table>
is that the number of solution steps is substantially reduced. This will be further investigated in the paper.
III. PROGRAM DEVELOPMENT AND USE

This section of the paper reports on program development and use. A complete program listing is included in the appendix. There is extensive internal documentation in this model. Due to this documentation and memory limitations this program can only handle line balancing problems of about fifty activities. With remark statements removed, the maximum number of activities increases to 250.

The program was written in Applesoft Basic. The goal was to develop a program in basic that would fit in the standard memory map of an Apple II series computer. Since Apple II series computers are built with basic language installed and with a minimum of 64K bytes of memory, this memory and language selection will make the program available for use by these computers.

Programming Consideration

Some special considerations that were included while developing this line balancing program are discussed here. These include: (1) the use of an activity matrix instead of a precedence matrix; (2) the control of precedence
numbers; (3) backtracking in the Hoffmann procedure; and
(4) automatic test generators.

Activity Matrix

The usual method of establishing the precedence requirements in a computer solution of a line balancing problem is with a precedence matrix. The development of a precedence matrix is discussed in the previous chapter. A precedence matrix requires a large amount of memory. For example, a program to solve a 100 element problem using a precedence matrix, would require a one 100 by 100 matrix. The dimension statement for this matrix causes an out of memory error on a microcomputer. To solve this problem an activity matrix is used. The first column of the matrix contains the preceding activities. The second column of the matrix contains the following activities. When developing precedence numbers, this method does have the disadvantage that it results in slower program operation. However, the microcomputer can now easily handle a 100 activity problem.

Precedence Numbers

Two key departures from the methods contained in the literature were required to establish and use precedence numbers. Initial precedence numbers were established by
searching the second column of the activity matrix. For every entry in the second column an activity's precedence number was increased by one.

Hoffmann (1963) maintained a matrix of precedence numbers for backtracking. This matrix was a "NA" by "NA" matrix, where "NA" is the number of activities. As mentioned above, using this method in a microcomputer would limit the number of activities that a program could solve. To solve this problem three sets of single column precedence matrixes were maintained. The Initial Precedence Number (IPN) Matrix was determined as described in the paragraph above. The other two sets of precedence number matrixes used were the Current Precedence Number (CPN) Matrix and the Test Precedence Number (TPN) Matrix. The CPN Matrix maintained the precedence numbers for the best solution at a station while backtracking was in progress. While searching activities for the best station assignment, the TPN Matrix was used. This matrix only had one column to save memory space.

Backtracking

Backtracking, as described by the Hoffmann procedure in the previous chapter, was accomplished by using three variables. These were: (1) a "K" variable that was increased by one every time a variable was added to a
station and decreased by one at every backtracking step; (2) the variable "KP(K)" was used as a pointer to record where to start a search of the "K'th" precedence numbers; (3) the TPN matrix described above is used to perform this backtracking.

Once the search reaches the point where no more activities meet the time and precedence requirements, then backtracking is performed. The steps used to perform the backtracking as used in the program are reproduced here.

1. Establish the previous TPN matrix by adding one to the precedence numbers of those activities following the activity last selected. Set the precedence number of the activity last selected to zero.
2. Add the time of the last activity selected to the time remaining at the station.
3. Remove the last activity from the list of activities to be assigned to the station.
4. Decrease the number of activities in the test assignment by one.
5. Decrease the KP(K) pointer by one.
6. Decrease the K pointer by one.
7. Continue the search for another test station assignment starting one element to the right of the last element selected.

Test Generators

Due to the relatively slow operation of a microcomputer, when solving line balancing problems, four automatic test generators were included in the program. These generators allow an operator to establish several different cycle times and delay factors for balances. The generators call up the proper subroutines to initiate the
requested balances. Since all output is sent to a printer
the operator is free to perform other tasks. These test
generators are described in more detail later in this
chapter.

Program Options and Use

Several options were built into the program to
provide a complete package for solving line balancing
problems. The program is menu driven and the operator is
prompted for the required inputs. Figure 2 shows the
main menu as it is displayed on the monitor. After the
execution of each of the program options, except the quit
option, the program returns to the main menu. Each of
the program options are described below.

Get Input Data from Disk

This option allows the operator to retrieve
previously stored activity matrix and activity times from
disk. When this option is selected the operator is
prompted for the file name. After the file name is
entered the subroutine retrieves data from the disk for
use in the program. The values retrieved are the project
name, number of activities, number of precedence
relationships, activity times, and precedence
relationships.
LINE BALANCING PROGRAM

PROGRAM OPTIONS

1. GET INPUT DATA FROM DISK.
2. GET INPUT DATA FROM KEYBOARD.
3. REVIEW/CHANGE ACTIVITIES.
4. PRINT ACTIVITIES.
5. STORE DATA ON DISK.
6. PERFORM HOFFMANN BALANCE.
7. PERFORM RPW BALANCE.
8. GET RESULTS FROM DISK.
9. DISPLAY RESULTS ON MONITOR.
10. PRINT RESULTS.
11. SAVE RESULTS ON DISK.
12. AUTO TEST GENERATOR.
13. QUIT.

ENTER OPTION

Figure 2. Main program menu showing the different program options.
Get Input Data from Keyboard

Activity times and precedence requirements are entered from the keyboard using this option. The operator is first asked for the project name. This project name can be any set of alphanumerics. After the project name is entered the operator is prompted for the number of activities. The program is initially dimensioned for fifty elements. When the proper number of activities has been established, instructions for entering precedence requirements are displayed. Activity times are entered after the precedence requirements.

Review/Change Activities

When this option is selected from the main menu, the activity times and precedence relationships are displayed. The operator is given an opportunity to change any of the times or precedence relationships.

Print Activities

In this subroutine activities currently in memory are reproduced on a printer. Output includes project name, precedence relationships, and activity times. The symbol ">>" means precedes. For example

1 >> 2

means activity one precedes activity two.
Store Data On Disk

After an assembly line's precedence relationships and activity times are entered they may be saved on disk using the option. The items save to disk are project name, number of activities, number of precedence relationships, activity times, and precedence relationships.

Perform Hoffmann Balance

A forward or reverse Hoffmann balance is performed by selecting this option. After selecting this option, a forward or reverse balance is selected. When the balance type is selected, other inputs to perform the balance are requested. These include number of stations (for a Modified Hoffmann balance), cycle time, and delay factor. If performing a Hoffmann balance without modifications, then enter one for the station number and zero for the delay factor. After the inputs are obtained, the balance is performed. During execution of the program, its progress in completing the balance is displayed on the monitor. When the balance is complete, the screen displays "balance complete," pauses and then returns to the main menu.

Perform RPW Balances

A forward or reverse Rank Position Weight balance is performed by selecting this option. After selecting this
option, a forward or reverse balance is selected. When the balance type is selected, a cycle time to perform the balance is requested. During program execution, its progress in completing the balance is displayed on the monitor. When the balance is complete the screen displays "balance complete," pauses and then returns to the main menu.

Get Results from Disk

This option allows results from balances stored on disk to be retrieved for display on a monitor or printer. After executing this option, the file name is requested. The values retrieved from disk are project name, balance type (forward or reverse), number of stations in the balance, number of activities, cycle time, standard deviation of the slack times, efficiency, total slack time, delay factor, balance mode (RPW or Hoffmann), station assignments, and station slack times.

Display Results on Monitor

When results of a balance are obtained, they can be displayed on a monitor by using this option. A sample output is shown in Figure 3. This output is for a forward Hoffmann balance of the problem shown in Figure 1 with a cycle time of fifteen. The output is
maintained on the screen for review until any key is depressed.

Print Results
This option sends balance results to the printer. The output is identical to that sent to the monitor. However, after the output is printed the program returns to the main menu without an operator action.

Save Results on Disk
This option allows results to be stored on disk. After selecting this option, the program asks for the file name where results will be stored. The values saved to disk are project name, balance type (forward or reverse), number of stations in the balance, number of activities, cycle time, standard deviation of the slack times, efficiency, total slack time, delay factor, balance mode (RPW or Hoffmann), station assignments, and station slack times.

Auto Test Generator
When this option is selected, the menu shown in Figure 4 appears. After the execution of each test generator the program sounds five bells and returns to the main menu. Each of the test generators will be discussed below.
MANSOOR 6
FORWARD HOFFMANN BALANCE

NUMBER OF ACTIVITIES 6
CYCLE TIME 15
MOD HOFFMANN DELAY FACTOR 0
NUMBER OF STATIONS 3
TOTAL SLACK TIME 5
STAND. DEV. OF SLACK TIME 1.25
LINE EFFICIENCY 88.89

<table>
<thead>
<tr>
<th>STATION NUMBER</th>
<th>ACTIVITIES ASSIGNED</th>
<th>SLACK TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>

PRESS ANY KEY TO CONTINUE LINE BALANCING PROGRAM

Figure 3. Sample program output showing a summary of the balance and station assignments.

AUTOMATIC TEST OPTIONS

1. GENERATE RPW BALANCES.
2. GENERATE HOFFMANN BALANCES.
3. BOTH 1 AND 2.
4. GENERATE MOD HOFFMANN BALANCES.

Figure 4. Auto test generator menu showing the different test generators available.
Generate RPW balances

This option generates forward and reverse RPW balances for up to ten different cycle times. Inputs obtained from the operator include number of different cycle times, cycle times, and a print option. The print option determines if individual station assignments are printed. All output is sent to a printer and includes project name, activity precedence relationships, activity times, and results of each balance.

Generate Hoffmann balances

This option generates forward and reverse Hoffmann balances with delay factors of zero, one-half, one, one and one-half, and two. Additionally, the operator may specify up to ten different cycle times. Inputs obtained from the operator include number of different cycle times, cycle times, and a print option. The print option determines if individual station assignments are printed. All output is sent to a printer and includes project name, activity precedence relationships, activity times, and results of each balance.

Both 1 and 2

This option generates all the balances described in the previous two options. As before, the operator can select up to ten different cycle times. Inputs obtained
from the operator include number of different cycle times, cycle times, and a print option. The print option determines if individual station assignments are printed. All output is sent to a printer and includes project name, activity precedence relationships, activity times, and results of each balance.

Generate Mod Hoffmann balances

This option generates modified Hoffmann balances for up to ten different cycle times and up to ten different delay factors. The number of stations used in determining the acceptable delay is determined by

$$[\text{ST}/\text{CT}]^+$$

where $\text{ST} = \text{Sum of all task times}$

$\text{CT} = \text{Cycle time}$

and $"[ \text{ } ]^+"$ means the next whole positive number. Inputs obtained from the operator include number of different cycle times, number of different delay factors, cycle times, delay factors, and a print option. The print option determines if individual station assignments are printed. All output is sent to a printer and includes project name, activity precedence relationships, activity times, and results of each balance.
Quit Option

Program operation is normally ended by executing this option. Prior to ending, a prompt is made to ensure that the operator wants to end the program. If he does, the program ends. Otherwise, operation is returned to the main menu. Another method to secure program operation is to hit "control-C." This will secure program operation anywhere in the program.
IV. TESTING AND RESULTS

To verify proper program operation and study different methods, several balances were performed. This section of the paper discusses testing and results.

Description of Problems Solved

A number of problems, ranging in size from six assembly task to forty-five assembly task, were taken from the literature to be used in this evaluation: Bedworth and Bailey (1982), Buffa (1969), Hoffmann (1963), Jackson (1956), Kilbridge and Wester (1961), Mansoor (1964), Moodie and Young (1965), and Tonge (1961). The cycle times used to assess performance were the same as those used in the literature. Additional times were used for the problems reported by Hoffmann, Moodie and Young, and Buffa since only one cycle time was reported in these articles. Thirty-seven balances were obtained using this literature data set over the range of cycle times considered.
Testing

A total of fourteen balances were conducted on each of the problems. This includes forward and reverse balances using both the Rank Positional Weight technique and the Hoffmann method. The Hoffmann method was solved using delay factors of 0.0, 0.5, 1.0, 1.25, 1.5, and 2.0. By considering fourteen different balancing schemes on thirty-seven different balancing problems, a total of 518 tests were conducted.

Results

Four measures of performance were considered in this study. The first two are standard comparisons found in the literature. The last two measures of performance used here were not noted in the literature. These measures of performance were: (1) the average percent a balance is above the optimum solution in terms of number of stations; (2) the time to obtain a balance; (3) the best solution in terms of the lowest number of station and lowest standard deviation of the slack times; and (4) the largest value for the minimum station slack time.

Average Percentage Above Optimum

Table four shows the average percentage increase above optimum solution for different methods studied and
compares these results to those obtained by Talbot, Patterson, and Gehrlein (1986), referred to here as the previous study. In general, this study found no difference between the performance of the different Hoffmann methods, in terms of number of stations only, until the delay factor increased to a value of two. This differs from the previous study in that the Hoffmann, without modifications, performed better than the Hoffmann with modifications. This difference occurs because the previous study considered some problems with a relatively large number of activities. Also, the previous study found the RPW method was worse than the Hoffmann method using a delay factor of two. This differs with the results of this study and is again accounted for by some of the balances studied. This is of little concern since both studies show that neither the RPW or Hoffmann 2.0 should be considered as a good solution method.

Time to Obtain a Balance

For a microcomputer the time to obtain a balance is a significant issue. The results of this study agree with those of the previous study in showing that the time to obtain a balance is reduced by using the modified Hoffmann procedure with a delay factor of at least one. This study found that on the average the Modified Hoffmann procedure,
with a delay factor of one, was two and one-half times faster than the Hoffmann procedure without modifications. This compares with 1.82 times faster in the previous study. It should also be noted that the difference in speed tended to increase as the number of activities increased. For example, using the program developed here it took seven minutes and thirty-two seconds for the Hoffmann $0.0$ to solve the Tonge (1961) twenty-one activity problem, with a cycle time of 20. The Hoffmann $1.0$ solution only took fifty-five seconds. Both obtained optimum solutions in terms of number of station. However, the Hoffmann $1.0$ solution had a lower standard deviation of slack times.

Standard Deviation of Slack Time

A comparison that was not noted in the literature was the performance when both the number of stations and the standard deviation of slack times are considered in selecting the best balance. When several methods give the same number of stations, then the balance that distributes the slack time evenly among stations should be selected. A measure of this distribution is the standard deviation of slack times. Table 5 shows the results that are obtained when both standard deviation of slack times and number of stations are considered. As can be seen, the
### Table 4

Comparison of the average percent above optimum performance in terms of number of stations

<table>
<thead>
<tr>
<th>METHOD</th>
<th>THIS STUDY</th>
<th>STUDY BY TALBOT, PATTERSON, GEHLEIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPW</td>
<td>3.11</td>
<td>4.62</td>
</tr>
<tr>
<td>HOFF 0.0</td>
<td>1.03</td>
<td>1.34</td>
</tr>
<tr>
<td>HOFF 0.5</td>
<td>1.03</td>
<td>3.29</td>
</tr>
<tr>
<td>HOFF 1.0</td>
<td>1.03</td>
<td>-</td>
</tr>
<tr>
<td>HOFF 1.25</td>
<td>1.03</td>
<td>-</td>
</tr>
<tr>
<td>HOFF 1.5</td>
<td>1.03</td>
<td>3.29</td>
</tr>
<tr>
<td>HOFF 2.0</td>
<td>3.63</td>
<td>3.41</td>
</tr>
</tbody>
</table>

### Table 5

Number of best solutions out of 37 balances when both number of stations and standard deviation of slack times are considered

<table>
<thead>
<tr>
<th>METHOD</th>
<th>NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPW</td>
<td>14</td>
</tr>
<tr>
<td>HOFF 0.0</td>
<td>20</td>
</tr>
<tr>
<td>HOFF 0.5</td>
<td>21</td>
</tr>
<tr>
<td>HOFF 1.0</td>
<td>24</td>
</tr>
<tr>
<td>HOFF 1.25</td>
<td>24</td>
</tr>
<tr>
<td>HOFF 1.5</td>
<td>30</td>
</tr>
<tr>
<td>HOFF 2.0</td>
<td>26</td>
</tr>
</tbody>
</table>
best results were obtained by Hoffmann 1.5. A delay factor of 1.5 allows slack times above and below the average slack time. The standard deviation is reduced since this allows a more even spread around the mean.

Increase in Minimum Slack Time

Another measure of performance that was not considered in the literature was the effect on minimum slack time when applying the Hoffmann method with delay factors. If a balance results in a minimum slack time greater than zero then the cycle time can be reduced by this minimum slack time and line efficiency improved. A study of the thirty-seven balances conducted revealed that on the average an increase in efficiency of 2.44% could be shown by applying this method with a delay factor of 1.5. The increase in efficiency was only 0.4% using Hoffmann without modifications. Most of the increase in efficiency was noted when large cycle times were used, as might be encountered when limited by the number of stations. This leads to a possible improved method of finding a balance for a Type II problem.

The usual method of finding the solution of a Type II problem is to start with the minimum possible cycle time and then increment this cycle time until a balance is
found. An alternate method that is suggested above is to solve the problem using the Modified Hoffmann procedure with a larger cycle time and then reduce the cycle time by the minimum slack time. This could result in fewer iterations and thus a quicker solution of the Type II problem. Developing and testing this method is an area where further study could be done.
V. CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK

This paper discussed the development, operation, and testing of a heuristic line balancing program that solves the Type I problem using a microcomputer. This program, which is included in the appendix, was written for use on a Apple II series computer.

The line balancing method selected was based on previous studies that compared different line balancing techniques. The model is built primarily using the Hoffmann (1963) procedure with modifications described by Gehrlein and Patterson (1975). For purposes of comparison, the Rank Positional Weight technique (Helgeson and Birnie 1961) is also included in the model.

Testing included thirty-seven different balances using problems from the literature. For each balance both Rank Positional Weight and Hoffmann solutions were obtained in the forward and reverse directions. Delay factors used for the Hoffmann balances were 0.0, 0.5, 1.0, 1.25, 1.5, and 2.0. Thus for each balance problem a total fourteen tests were conducted. The total number of tests was 518.
Four measures of performance were considered in this study. These measures of performance were: (1) the average percentage a balance is above the optimum solution, in terms of number of stations; (2) time to obtain a balance; (3) the best solution in terms of the lowest number of stations and lowest standard deviation of the slack times, and (4) the largest value of minimum station slack time. Overall it was found that the Modified Hoffmann procedure with a delay factor of 1.5 performed best. The time to obtain a balance using the Modified Hoffmann procedure is significantly less than the time for a solution without modifications.

Two areas of further work are suggested by this paper. The first is to conduct a more in-depth study to find the best delay factor when using the Modified Hoffmann procedure on a microcomputer. The variation of the best delay factor could be studied to determine how it varies in terms of problem size and cycle time. Another area where future study can be directed is to develop and test the method for solving Type II problems suggested by using a Modified Hoffmann procedure, with a cycle time sufficiently large to generate slack time at each station, and then reduce the cycle time by the minimum slack time. A faster solution of Type II problems will result if the number of iterations required is reduced.
APPENDIX

BALANCE PROGRAM LISTING
100 REM THIS PROGRAM CONTAINS TWO
105 REM LINE BALANCING METHODS
110 REM
115 REM THE TWO METHODS ARE THE
120 REM RANK POSITION WEIGHT METHOD
125 REM AND THE HOFFMANN METHOD
130 REM
135 REM THE RANK POSITION WEIGHT
140 REM METHOD IS REFERENCED IN
145 REM J. IND. ENG., 12, 6
150 REM (NOV-DEC, 1961), P394-398.
155 REM HELGESON, W.P. AND
160 REM D.P. BIRNIE
165 REM
170 REM THE HOFFMANN METHOD IS
175 REM REFERENCED IN
180 REM MAN. SCIENCE, 9, 4 (JULY, 1963)
185 REM P551-562.
190 REM THE BASIC HOFFMANN METHOD
195 REM IS ALSO MODIFIED HERE
200 REM AS DESCRIBED BY
205 REM GEHRLIN, W. V. AND
210 REM H. PATTERSON IN
215 REM MAN. SCIENCE, 21, 9 (MAY 1975)
220 REM P1064-1070.
225 REM
230 REM INSTEAD OF USING A PRECEDENCE
235 REM MATRIX WHICH WOULD TAKE UP TO
240 REM MUCH MEMORY AN ACTIVITY MATRIX
245 REM IS USED. THE FIRST COLUMN IN
250 REM THE ACTIVITY MATRIX CONTAINS
255 REM THE PRECEDING ACTIVITIES.
260 REM THE FOLLOWING ACTIVITIES
265 REM ARE IN THE SECOND COLUMN. AN
270 REM ADDITIONAL CHANGE FROM THE
275 REM HOFFMANN PROCEDURE IS THAT
280 REM A PRECEDENCE NUMBER IS
285 REM MAINTAINED FOR EACH ACTIVITY
290 REM VICE A MATRIX OF ALL "K NUMBERS"
295 REM THE CURRENT PRECEDENCE NUMBERS
300 REM ARE USED AND RECALCULATED
305 REM WHEN REQUIRED.
310 REM
315 REM THE VARIABLES USED IN THIS MODEL
320 REM AND THEIR MEANING ARE DISCUSSED
325 REM BELOW.
330 REM AA(I)
335 REM MATRIX WHOSE VALUE IS THE
340 REM STATION ASSIGNMENT OF
345 REM ACTIVITY I
350 REM
355 REM AD
360 REM THE ACCEPTABLE DELAY
365 REM DETERMINED FOR THE MODIFIED
370 REM HOFFMANN BALANCE.
375 REM
380 REM AP
385 REM DESIGNATES THE POSITION OF
390 REM THE FOLLOWING ACTIVITY IN
395 REM A BALANCE. IF IT IS A
400 REM FORWARD BALANCE THEN AF = 2
405 REM FOR A REVERSED BALANCE THE
410 REM AF = 1
415 REM
420 REM
425 REM AI
430 REM COUNTER IN FOR NEXT STATEMENT
435 REM
440 REM AJ
445 REM COUNTER IN FOR NEXT STATEMENT
450 REM
455 REM AM(I,J)
460 REM ACTIVITY MATRIX THAT INDICATES
465 REM THE PRECEDENCE REQUIREMENTS
470 REM FOR THE VARIABLES.
475 REM
480 REM AP
485 REM DESIGNATES THE POSITION OF
490 REM THE PRECEDING ACTIVITY IN
495 REM A BALANCE. IF IT IS A
500 REM FORWARD BALANCE THEN AP = 1.
505 REM FOR A REVERSED BALANCE THEN
510 REM AP = 2.
515 REM
520 REM AS
525 REM ACTIVITY SELECTION POINTER
530 REM AS = 1 IMPLIES ACTIVITY SELECTED
535 REM AS = 0 IMPLIES ACTIVITY NOT SELECTED
540 REM
545 REM AU
550 REM POINTER THAT INDICATES WHEN
555 REM AN AUTOMATIC TEST GENERATOR
560 REM IS DRIVING THE PROGRAM.
AU = 1 IMPLIES TEST GENERATOR OPERATING. AU = 0 IMPLIES TEST GENERATOR NOT OPERATING

VARIABLE B$ CONTAINS THE CONTROL CHARACTER FOR AN AUDIBLE BELL

BC POINTER THAT INDICATES WHEN THE BALANCE IS COMPLETE

BC = 1 IMPLIES BALANCE COMPLETE

BC = 0 IMPLIES BALANCE NOT COMPLETE

BM POINTER THAT INDICATES BALANCE TYPE. BM = 1 IMPLIES RPW BALANCE

BM = 2 IMPLIES HOFFMANN BALANCE

BT POINTER THAT INDICATES BALANCE TYPE. BT = 1 IMPLIES FORWARD BALANCE.

BT = -1 IMPLIES REVERSE BALANCE.

THE TOTAL NUMBER OF PRECEDENCE REQUIREMENTS IN THE ACTIVITY MATRIX (IE. # OF COLUMNS IN THE ACTIVITY MATRIX)

THE NUMBER OF DIFFERENT CYCLE TIMES THAT WILL BE USED WHEN OPERATING THE AUTOMATIC TEST GENERATOR.

THE NUMBER OF ACTIVITIES ASSIGNED TO THE CURRENT STATION SELECTION. USED IN THE HOFFMANN PROCEDURE.

CURRENT PRECEDENCE NUMBER FOR ACTIVITY I BASED ON THE CURRENT STATION ASSIGNMENT.
CRPW
CURRENT VALUE OF THE RANK
POSITION WEIGHT OF THE
BEST TASKS FOR ASSIGNMENT
TO THE STATION.
REM
CSA(I) C
CURRENT STATION ASSIGNMENT
VALUE IN MATRIX IS THE
ACTIVITY NUMBER THAT IS
ASSIGNED TO A STATION.
REM
CT
REM
CT(I)
REM
CT(I)
MATRIX OF CYCLE TIMES
USED IN AUTOMATIC TEST
GENERATORS.
REM
D$
REM CONTAINS THE VALUE OF
CONTROL D
REM
DD
NUMBER OF DIFFERENT
HOFFMANN DELAY FACTORS
USED IN A TEST GENERATOR.
REM
DF
HOFFMANN DELAY FACTOR
REM
DF(I)
MATRIX OF DELAY FACTORS
USED IN AUTOMATIC TEST
GENERATORS.
REM
EFF
EFFICIENCY OF THE BALANCE
IN PERCENT.
REM
EP
ENDING POINT OF A SEARCH
FOR AN ACTIVITY FOR A
GIVEN BALANCE TYPE (FORWARD
OR REVERSE). EP = NA FOR A
FORWARD BALANCE. EP = 1 FOR
A REVERSE BALANCE.

FILE NAME
COUNTERS USED IN FOR-NEXT STATEMENTS
IE
THE ENDING POINT WHEN PRINTING DATA TO THE MONITOR.
THE INITIAL PRECEDENCE NUMBER
FOR ACTIVITY I
FOR ACTIVITY I
IS
THE STARTING POINT OF A PRINTING ROUTINE
J
COUNTER USED IN FOR-NEXT STATEMENTS
K
THE CURRENT VALUE OF THE K#
USED IN THE HOFFMANN BALANCE
K$
A DUMMY VARIABLE FOR PROGRAM CONTROL
KP(I)
A POINTER USED IN THE BACKTRACKING ROUTINE IN THE HOFFMANN BALANCE
TO KEEP TRACK OF WHERE THE SEARCH SHOULD START.
L,M
COUNTERS USED IN FOR-NEXT STATEMENTS.
MNP
THE MINIMUM NUMBER OF STATIONS POSSIBLE GIVEN THE PROBLEM SIZE AND CYCLE TIME
NA
1265 REM NUMBER OF ACTIVITIES
1270 REM
1275 REM NR
1280 REM NUMBER ROUNDER USED FOR
1285 REM PRINTING RESULTS
1290 REM
1295 REM NS
1300 REM NUMBER OF STATIONS PREDICTOR
1305 REM USED IN THE MODIFIED HOFFMANN
1310 REM PROCEDURE
1315 REM
1320 REM OP
1325 REM OPTION CODE THAT SELECTS
1330 REM OPTIONS FROM A MENU
1335 REM
1340 REM P$
1345 REM CONTAINS THE PRODOS BASIC
1350 REM PREFIX FOR SAVING OR
1355 REM RETREIVING DATA TO DISK.
1360 REM
1365 REM PA
1370 REM POINTER THAT CONTAINS A
1375 REM PRINT OPTION TO PRINT
1380 REM STATION ASSIGNMENTS.
1385 REM PA = 1 IMPLIES PRINT STATION
1390 REM ASSIGNMENTS. PA = 0 IMPLIES
1395 REM DO NOT PRINT STATION ASSIGNMENTS.
1400 REM
1405 REM PC
1410 REM POINTER THAT INDICATES IF
1415 REM THE INITIAL PRECEDENCE NUMBERS
1420 REM AND RANK POSITION WEIGHTS
1425 REM HAVE BEEN DETERMINED.
1430 REM
1435 REM PN$
1440 REM PROJECT NAME
1445 REM
1450 REM R$
1455 REM CONTAINS THE PRODOS PREFIX FOR
1460 REM SAVING OR RETREIVING RESULTS.
1465 REM
1470 REM RPW(I)
1475 REM RANK POSITION WEIGHT FOR ACTIVITY I
1480 REM
1485 REM RT(I)
1490 REM INTERMEDIATE VARIABLE TO
1495 REM REVERSE THE ORDER OF THE
SLACK TIMES IN A REVERSE BALANCE.

STARTING ACTIVITY FOR A GIVEN K NUMBER IN SEARCHING TO SELECT AN ACTIVITY. USED IN THE HOFFMANN.

PROCEDURE.

SC

STARTING ACTIVITY FOR A GIVEN K NUMBER IN SEARCHING TO SELECT AN ACTIVITY. USED IN THE HOFFMANN.

SD

STANDARD DEVIATION OF THE SLACK TIMES.

SEL

NUMBER OF THE ACTIVITY SELECTED FOR ASSIGNMENT TO A STATION.

SEL

NUMBER OF THE ACTIVITY SELECTED FOR ASSIGNMENT TO A STATION.

SF

SCREEN FULL POINTER. KEEPS THE NUMBER OF LINES ON A SCREEN.

SN

STATION NUMBER. AT THE END OF THE BALANCE IT EQUALS THE NUMBER OF STATIONS REQUIRED IN THE BALANCE. DURING THE BALANCE IT IS THE CURRENT STATION NUMBER.

SF

SCREEN FULL POINTER. KEEPS THE NUMBER OF LINES ON A SCREEN.

SN

STATION NUMBER. AT THE END OF THE BALANCE IT EQUALS THE NUMBER OF STATIONS REQUIRED IN THE BALANCE. DURING THE STANDARD DEVIATION OF THE SLACK TIMES.

SEL

NUMBER OF THE ACTIVITY SELECTED FOR ASSIGNMENT TO A STATION.

SF

SCREEN FULL POINTER. KEEPS THE NUMBER OF LINES ON A SCREEN.

SN

STATION NUMBER. AT THE END OF THE BALANCE IT EQUALS THE NUMBER OF STATIONS REQUIRED IN THE BALANCE. DURING THE STANDARD DEVIATION OF THE SLACK TIMES.

SEL

NUMBER OF THE ACTIVITY SELECTED FOR ASSIGNMENT TO A STATION.

SF

SCREEN FULL POINTER. KEEPS THE NUMBER OF LINES ON A SCREEN.

SN

STATION NUMBER. AT THE END OF THE BALANCE IT EQUALS THE NUMBER OF STATIONS REQUIRED IN THE BALANCE. DURING THE STANDARD DEVIATION OF THE SLACK TIMES.

SEL

NUMBER OF THE ACTIVITY SELECTED FOR ASSIGNMENT TO A STATION.
REM TL(I) SLACK TIME AT STATION I
REM TN
REM NUMBER OF ACTIVITIES IN THE CURRENT STATION TEST SOLUTION. USED IN THE HOFFMANN PROCEDURE.
REM TP
REM POINTER USED THE PRINT SUB IF THE STATION SLACK TIME HAS BEEN PRINTED.
REM TP = 0 IMPLIES STATION SLACK TIME NOT PRINTED.
REM TP = 1 IMPLIES STATION SLACK TIME HAS BEEN PRINTED.
REM TP = 0 IMPLIES STATION SLACK TIME NOT PRINTED.
REM TP = 1 IMPLIES STATION SLACK TIME HAS BEEN PRINTED.
REM TPN(I) PRECEDENCE NUMBER FOR THE CURRENT TEST SOLUTION. USED IN THE HOFFMANN PROCEDURE.
REM TRPW TEST RANK POSITION WEIGHT FOR THE ACTIVITY BEING TESTED FOR ASSIGNMENT.
REM TSA(I) TEST STATION ASSIGNMENT FOR THE CURRENT TEST SOLUTION.
REM TTL TIME LEFT IN THE CURRENT TEST ASSIGNMENT OF TASK T(I).
REM Y OR N VARIABLE FOR PROGRAM CONTROL.
REM START OF MAIN MENU.
1970 REM START OF MAIN MENU
1975 REM
1980 B$ = CHR$ (7):
    REM BELL
1985 D$ = CHR$ (4):
    REM CONTROL D
1990 P$ = "\HOFFMANN\DATA/":
R$ = "\HOFFMANN\RESULTS/"
1995 PA = 1
2000 DIM CPN(50):
    DIM TPN(50):
    DIM IPN(50)
2005 DIM AM(100,2)
2010 DIM CSA(20):
    DIM TSA(20):
    DIM KP(20)
2015 DIM T(50):
    DIM RPW(50)
2020 DIM TL(50):
    DIM CT(10)
2025 DIM RT(50):
    DIM DF(10)
2030 DIM AA(50)
2035 HOME
2040 PRINT "LINE BALANCING PROGRAM"
2045 REM
2050 REM LIST PROGRAM OPTIONS
2055 REM
2060 PRINT:
    PRINT "PROGRAM OPTIONS"
2065 PRINT:
    PRINT
2070 PRINT TAB(5)"1. GET INPUT DATA FROM DISK."
2075 PRINT TAB(5)"2. GET INPUT DATA FROM KEYBOARD."
2080 PRINT TAB(5)"3. REVIEW/CHANGE ACTIVITIES."
2085 PRINT TAB(5)"4. PRINT ACTIVITIES."
2090 PRINT TAB(5)"5. STORE DATA ON DISK."
2095 PRINT TAB(5)"6. PERFORM HOFFMANN BALANCE."
2100 PRINT TAB(5)"7. PERFORM RPW BALANCE."
2105 PRINT TAB(5)"8. GET RESULTS FROM DISK."
2110 PRINT TAB(5)"9. DISPLAY RESULTS ON MONITOR."
2115 PRINT TAB(4)"10. PRINT RESULTS."
2120 PRINT TAB(4)"11. SAVE RESULTS ON DISK."
2125 PRINT TAB(4)"12. AUTO TEST GENERATOR."
2130 PRINT TAB(4)"13. QUIT."
2135 PRINT
2140 INPUT "ENTER OPTION ";OP
2145 IF (OP > 0) AND (OP < 13) THEN 2180
2150 PRINT B$
2155 PRINT "INVALID INPUT"
2160 PRINT "ENTER OPTION AS A"
2165 PRINT "NUMBER 1 TO 13."
2170 FOR ID = 1 TO 1000: NEXT ID
2175 GOTO 2035
2180 ON OP GOSUB 2190, 2290, 4450, 3820, 2510,
4820, 4920, 3970, 3770, 3725, 4070, 5025, 4185
2185 GOTO 2035
2190 REM
2195 REM SUBROUTINE TO GET DATA
2200 REM FROM DISK
2205 REM
2210 PRINT :
2215 PRINT "ENTER NAME OF FILE ";F$
2220 PRINT D$;"OPEN";P$;F$
2225 PRINT D$;"READ";P$;F$
2225 INPUT PN$:
2230 REM GET ACTIVITY TIMES
2235 FOR I = 1 TO NA
2240 INPUT T(I)
2245 NEXT I
2250 REM GET ACTIVITY MATRIX
2255 FOR J = 1 TO 2
2260 FOR I = 1 TO C
2265 INPUT AM(I,J)
2270 NEXT I
2275 NEXT J
2280 PRINT D$;"CLOSE";P$;F$
2285 RETURN
2290 REM
2295 REM SUBROUTINE TO GET INPUT DATA
2300 REM FROM KEYBOARD
2305 REM
2310 PRINT :
2315 PRINT "ENTER PROJECT NAME ";PN$
2320 PRINT
2325 PRINT "ENTER THE NUMBER OF"
2330 INPUT "ACTIVITIES(1 - 50) ";NA
2335 PRINT
2340 IF (NA > = 1) AND (NA < = 50) THEN 2370
2345 PRINT :
PRINT "INVALID ENTRY"
2350 PRINT "NUMBER OF ACTIVITIES MUST BE"
2355 PRINT "BETWEEN 1 AND 50"
2360 PRINT
2365 GOTO 2320
2370 REM DATA INPUT
2375 C = 1
2380 PRINT "ENTER ACTIVITY PRECEDENCE"
2385 PRINT "DATA AS PRECEDING ACTIVITY"
2390 PRINT "NUMBER, FOLLOWING ACTIVITY"
2395 PRINT "NUMBER."
2400 PRINT "EXAMPLE: IF ACTIVITY TWO"
2405 PRINT "IMMEDIATELY FOLLOWS ACTIVITY"
2410 PRINT "ONE THE ENTER AS 1,2."
2415 PRINT "WHEN COMPLETE ENTER 0,0."
2420 PRINT
2425 PRINT "A PRECEDES B"
2430 SF = 1
2435 PRINT
2440 INPUT "A,B ";AM(C,1),AM(C,2)
2445 IF (AM(C,1) = 0) AND (AM(C,2) = 0) THEN 2470
2450 C = C + 1
2455 IF SF = 10 THEN 2380
2460 SF = SF + 1
2465 GOTO 2440
2470 C = C - 1
2475 PRINT :
PRINT "ENTER ACTIVITY TIMES."
2480 PRINT
2485 FOR I = 1 TO NA
2490 PRINT "ENTER TIME FOR ACTIVITY ";I;
2495 INPUT " ";T(I)
2500 NEXT I
2505 RETURN
2510 REM
2515 REM SUBROUTINE TO STORE DATA ON
2520 REM DISK
2525 REM
2530 INPUT "ENTER FILE NAME ";F$
2535 PRINT D$;"OPEN";P$;F$
2540 PRINT D$;"CLOSE";P$;F$
2545 PRINT D$;"DELETE";P$;F$
2550 PRINT D$;"OPEN";P$;F$
2555 PRINT D$;"WRITE";P$;F$
2560 PRINT PN$:
PRINT NA:
PRINT C
2565 REM SAVE ACTIVITY TIMES
2570 FOR I = 1 TO NA
2575 PRINT T(I)
2580 NEXT I
2585 REM SAVE ACTIVITY MATRIX
2590 FOR J = 1 TO 2
2595 FOR I = 1 TO C
2600 PRINT AM(I,J)
2605 NEXT I
2610 NEXT J
2615 PRINT D$;"CLOSE";P$;F$
2620 RETURN
2625 REM
2630 REM FORWARD HOFFMANN SUBROUTINE
2635 REM
2640 REM ESTABLISH VALUES FOR A FORWARD BALANCE
2645 REM
2650 PRINT :
   PRINT :
   PRINT :
2655 GOSUB 2870:
   PRINT :
   PRINT :
2670 REM ACCEPTANCE DELAY SUBROUTINE
2660 AP = 1
2665 BT = 1:
2670 BM = 2
2675 EP = NA
2675 SP = 1
2680 AF = 2
2685 PRINT :
   PRINT :
2690 PRINT "CYCLE TIME ",CT
2695 PRINT "DELAY FACTOR ";DF
2700 PRINT "PERFORMING FORWARD HOFFMANN BALANCE"
2705 PRINT :
   PRINT :
2710 IF PC = 1 THEN 2720
2715 GOSUB 2965:
   PRINT :
   PRINT :
2720 GOSUB 3020:
   PRINT :
2725 PRINT "BALANCE COMPLETE"
2730 PRINT :
   PRINT :
2735 FOR M = 1 TO 1000:
   NEXT M
2740 RETURN
2745 REM
2750 REM REVERSE HOFFMANN SUBROUTINE
2755 REM
2760 REM ESTABLISH VALUES FOR A REVERSE BALANCE
2765 REM
2770 PRINT:
PRINT:
PRINT:
2775 GOSUB 2870:
REM ACCEPTANCE DELAY SUBROUTINE
2780 BM = 2
2785 AF = 1:
AP = 2:
BT = - 1:
EP = 1:
SP = NA
2790 PRINT:
PRINT:
PRINT:
2795 PRINT "CYCLE TIME ";CT
2800 PRINT "DELAY FACTOR ";DF
2805 PRINT "PERFORMING REVERSE HOFFMANN BALANCE"
2810 PRINT:
PRINT:
PRINT:
2815 IF PC = 1 THEN 2825
2820 GOSUB 2965:
REM DETERMINE PRECEDENCE NUMBERS
2825 GOSUB 3020
2830 REM REVERSE THE ORDER OF STATION ASSIGNMENTS
2835 FOR I = 1 TO NA:
AA(I) = SN - AA(I) + 1:
NEXT I
2840 REM REVERSE THE ORDER OF SLACK TIME ASSIGNMENTS
2845 FOR I = 1 TO SN:
RT(I) = TL(SN - I + 1):
NEXT I
2850 FOR I = 1 TO SN:
TL(I) = RT(I):
NEXT I
2855 PRINT:
PRINT:
PRINT:
PRINT "BALANCE COMPLETE"
2860 FOR I = 1 TO 1000:
NEXT I
RETURN
REM ACCEPTABLE DELAY SUBROUTINE
REM DETERMINE TOTAL TASK TIME
ST = 0
FOR I = 1 TO NA:
ST = ST + T(I):
NEXT I
IF AU = 1 THEN 2945
REM GET INPUTS FOR BALANCE
PRINT:
PRINT "ENTER NUMBER OF STATIONS"
INPUT "FOR HOFFMANN 0.0 ENTER 1 "; NS
INPUT "CYCLE TIME "; CT
PRINT "ENTER DELAY FACTOR"
INPUT "FOR HOFFMANN 0.0 ENTER 0 "; DF
REM DETERMINE ACCEPTABLE STATION DELAY
AD = DF * ((CT * NS - ST) / NS)
RETURN
REM DETERMINE INITIAL PRECEDENCE NUMBERS SUBROUTINE
REM INITIALIZE VARIABLES
FOR I = 1 TO 20:
CSA(I) = 0:
TSA(I) = 0:
KP(I) = SP - BT:
NEXT I
3065 TL(SN) = CT:
TTL = CT:
TN = 0:
CN = 0:
K = 1
3070 PRINT "PERFORMING STATION ";SN;" BALANCE."
3075 SC = KP(K) + BT:
    REM STARTING POINT OF EACH COMBINATION
3080 FOR I = SC TO EP STEP BT:
    REM START BALANCE
3085 KP(K) = I
3090 IF TPN(I) < > 0 THEN 3160
3095 IF T(I) > TTL THEN 3160
3100 TN = TN + 1:
    REM ACTIVITY ADDED TO TEST SELECTION
3105 TSA(TN) = I
3110 TTL = TTL - T(I)
3115 K = K + 1:
    REM INCREMENT K NUMBER
3120 REM ESTABLISH NEW TEST PRECEDENCE NUMBERS
3125 FOR J = 1 TO C
3130 IF AM(J,AP) < > I THEN 3140
3135 TPN(AM(J,AP)) = TPN(AM(J,AP)) - 1
3140 NEXT J
3145 TPN(I) = - 10
3150 IF TTL < = AD THEN 3175
3155 GOTO 3075
3160 NEXT I:
    REM SELECTION OF NEW COMBINATION COMPLETE
3165 REM TEST FOR ACCEPTANCE OF NEW COMBINATION
3170 IF TTL > = TL(SN) THEN 3250
3175 TL(SN) = TTL:
    CN = TN
3180 FOR J = 1 TO TN:
    REM SET CURRENT STATION ASSIGNMENTS
3185 CSA(J) = TSA(J):
    REM EQUAL TO TEST STATION ASSIGNMENTS
3190 NEXT J
3195 REM SET CURRENT PRECEDENCE NUMBERS
3200 REM EQUAL TO TEST PRECEDENCE NUMBERS
3205 FOR J = 1 TO NA:
    CPN(J) = TPN(J):
    NEXT J
3210 REM TEST FOR ALL CPN(I) EQUAL TO -10
3215 FOR L = 1 TO NA
3220 IF CPN(L) = - 10 THEN 3230
3225 GOTO 3245
3230  NEXT L
3235  BC = 1:
   REM  BALANCE COMPLETE
3240  GOTO 3340
3245  IF TTL < = AD THEN 3340:
   REM  STATION SELECTION COMPLETE
3250  IF KP(1) = EP THEN 3340
3255  REM
3260  REM  BACKTRACK. ESTABLISH BACK VALUES FOR
3265  REM  NEW TPN(I), TTL, TSA, TN, KP(I), K
3270  REM
3275  FOR L = 1 TO C
3280  IF AM(L, AP) < > TSA(TN) THEN 3290
3285  TPN(AM(L, AF)) = TPN(AM(L, AF)) + 1
3290  NEXT L
3295  TPN(TSA(TN)) = 0
3300  TTL = TTL + T(TSA(TN))
3305  TSA(TN) = 0
3310  TN = TN - 1
3315  KP(K) = SP - BT
3320  K = K - 1
3325  IF KP(K) = EP THEN 3250
3330  GOTO 3075
3335  REM
3340  REM  RESULTS FINAL FOR STATION
3345  REM  ASSIGN ACTIVITIES
3350  REM  SETUP TO SELECT ACTIVITIES
3355  REM  FOR NEXT STATION
3360  REM
3365  FOR M = 1 TO CN:
   AA(CSA(M)) = SN
3370  PRINT "ACTIVITY ";CSA(M);" TO STATION ";SN
3375  NEXT M
3380  IF BC = 1 THEN 3405:
   REM  BALANCE COMPLETE
3385  FOR M = 1 TO NA:
   TPN(M) = CPN(M):
      NEXT M
3390  SN = SN + 1
3395  GOTO 3060
3400  REM
3405  REM  BALANCE COMPLETE
3410  REM  DETERMINE STANDARD DEVIATION FOR 
3415  REM  SLACK TIMES
3420  REM
3425  SU = 0:
   SS = 0
3430 FOR I = 1 TO SN
3435 SU = SU + TL(I)
3440 SS = SS + TL(I) * TL(I)
3445 NEXT I
3450 SD = ((SS - (SU * SU) / SN)) / SN) ^ 0.5
3455 REM
3460 REM DETERMINE LINE EFFICIENCY
3465 REM
3470 ST = 0
3475 FOR I = 1 TO NA:
3480 ST = ST + T(I):
3485 NEXT I
3490 EFF = (ST / (CT * SN)) * 100
3495 RETURN
3500 REM
3505 REM SUBROUTINE TO DISPLAY RESULTS
3510 PRINT PN$
3515 IF (BT = 1) AND (BM = 1) THEN 3535
3520 IF (BT = 1) AND (BM = 2) THEN 3540
3525 IF (BM = 1) THEN 3545
3530 GOTO 3550
3535 PRINT "FORWARD RPW BALANCE":
3540 PRINT "FORWARD HOFFMANN BALANCE":
3545 PRINT "REVERSE RPW BALANCE":
3550 PRINT "REVERSE HOFFMANN BALANCE":
3555 PRINT "NUMBER OF ACTIVITIES ";NA
3560 PRINT "CYCLE TIME ";CT
3565 IF BM = 1 THEN 3575
3570 PRINT "MOD HOFFMANN DELAY FACTOR ";DF
3575 PRINT "NUMBER OF STATIONS ";SN
3580 PRINT "TOTAL SLACK TIME ";SU
3585 NR = ( INT (100 * SD + 0.5)) / 100
3590 PRINT "STAND. DEV. OF SLACK TIME ";NR
3595 NR = ( INT (100 * EFF + 0.5)) / 100
3600 PRINT "LINE EFFICIENCY ";NR
3605 PRINT
3610 IF PA = 0 THEN 3720
3615  HTAB (5):
      PRINT "STATION"
3620  HTAB (20):
      PRINT "ACTIVITIES"
3625  HTAB (35):
      PRINT "SLACK"
3630  HTAB (5):
      PRINT "NUMBER"
3635  HTAB (20):
      PRINT "ASSIGNED"
3640  HTAB (35):
      PRINT "TIME"
3645  PRINT
3650  FOR I = 1 TO SN
3655  TP = 0
3660  PRINT
3665  FOR J = 1 TO NA
3670  IF AA(J) < > I THEN 3710
3675  HTAB (5):
      PRINT I;
3680  IF TP < > 0 THEN 3705
3685  HTAB (20):
      PRINT J;
3690  HTAB (35):
      PRINT TL(I)
3695  TP = 1
3700  GOTO 3710
3705  HTAB (20):
      PRINT J
3710  NEXT J
3715  NEXT I
3720  RETURN
3725  REM
3730  REM PRINT RESULTS SUBROUTINE
3735  REM
3740  PRINT D$;"PR#1"
3745  PRINT :
      PRINT
3750  GOSUB 3490
3755  PRINT :
      PRINT
3760  PRINT D$;"PR#0"
3765  RETURN
3770  REM
3775  REM DISPLAY RESULTS ON
REM MONITOR SUBROUTINE

GOSUB 3490
PRINT "PRESS ANY KEY TO CONTINUE";
GET K$
RETURN

REM SUBROUTINE TO PRINT ACTIVITIES
PRINT D$;'PR#1'
PRINT :
PRINT
PRINT PN$
PRINT
HTAB (5): PRINT "ACTIVITY";
HTAB (15): PRINT "PRECEDES";
HTAB (25): PRINT "ACTIVITY"
HTAB (5): PRINT "A";
HTAB (25): PRINT "B"
PRINT FOR I = 1 TO C
HTAB (5): PRINT AM(I,1);
HTAB (15): PRINT ">>";
HTAB (25): PRINT AM(I,2)
NEXT I
PRINT
REM PRINT ACTIVITY TIMES
HTAB (5): PRINT "ACTIVITY";
HTAB (15): PRINT "TIME"
PRINT FOR I = 1 TO NA
HTAB (5): PRINT I;
HTAB (15): PRINT T(I)
NEXT I
PRINT : PRINT : PRINT
PRINT D$;"PR#0"
RETURN
REM
REM SUB ROUTINE TO GET
REM RESULTS FROM DISK
REM
PRINT : PRINT
PRINT "ENTER NAME OF FILE ";F$
PRINT D$;"OPEN";R$;F$
PRINT D$;"READ";R$;F$
INPUT PN$:
INPUT BT:
INPUT SN
INPUT NA:
INPUT CT:
INPUT SD
INPUT EFF:
INPUT SU:
INPUT DF
INPUT BM
FOR I = 1 TO NA
INPUT AA(I)
NEXT I
FOR I = 1 TO SN
INPUT TL(I)
NEXT I
PRINT D$;"CLOSE";R$;F$
RETURN
REM
REM SUBROUTINE TO SAVE
REM RESULTS ON DISK
REM
PRINT : PRINT
PRINT "ENTER FILE NAME ";F$
PRINT D$;"OPEN";R$;F$
PRINT D$;"CLOSE";R$;F$
PRINT D$;"DELETE";R$;F$
PRINT D$;"OPEN";R$;F$
PRINT D$;"WRITE";R$;F$
PRINT PN$:
PRINT BT:
PRINT SN
4130 PRINT NA:
  PRINT CT:
  PRINT SD
4135 PRINT EFF:
  PRINT SU:
  PRINT DF
4140 INPUT BM
4145 FOR I = 1 TO NA
4150 PRINT AA(I)
4155 NEXT I
4160 FOR I = 1 TO SN
4165 PRINT TL(I)
4170 NEXT I
4175 PRINT D$;"CLOSE";R$;F$
4180 RETURN
4185 REM
4190 REM QUIT SUBROUTINE
4195 REM
4200 PRINT:
  PRINT:
  PRINT
4205 PRINT "YOU ARE ABOUT TO END THE PROGRAM."
4210 PRINT "IS THIS WHAT YOU WANT TO DO?"
4215 INPUT "ENTER Y OR N ";YN$
4220 IF YN$ = "Y" THEN 4230
4225 RETURN
4230 END
4235 REM
4240 REM AUTO TEST GENERATOR SUBROUTINE
4245 REM THAT WILL GENERATE FORWARD AND REVERSE HOFFMANN
4250 REM BALANCES WITH DELAY FACTORS FROM
4255 REM ZERO TO TWO IN INCREMENTS OF 0.5.
4260 REM
4265 AU = 1
4270 PRINT "ENTER THE NUMBER OF DIFFERENT"
4275 INPUT "CYCLE TIMES THAT WILL BE USED ";CC
4280 FOR I = 1 TO CC
4285 PRINT "ENTER CYCLE TIME ";I;
4290 INPUT CT(I)
4295 NEXT I
4300 PRINT "DO YOU WANT TO PRINT STATION"
4305 INPUT "ASSIGNMENTS? ENTER Y OR N. ";YN$
4310 PA = 1
4315 IF YN$ = "N" THEN PA = 0
4320 GOSUB 3820
4325 FOR AI = 1 TO CC
4330 CT = CT(AI)
4335 DF = 0:
NS = 1:
PC = 0
4340 GOSUB 2625
4345 IF DF > 0 THEN 4355
4350 NS = SN
4355 GOSUB 3725
4360 IF DF = 2 THEN 4380
4365 DF = DF + 0.5
4370 PC = 1
4375 GOTO 4340
4380 DF = 0:
NS = 1:
PC = 0
4385 GOSUB 2745
4390 IF DF > 0 THEN 4400
4395 NS = SN
4400 GOSUB 3725
4405 IF DF = 2 THEN 4425
4410 DF = DF + 0.5
4415 PC = 1
4420 GOTO 4385
4425 NEXT AI
4430 AU = 0:
PA = 1:
PC = 0
4435 FOR IB = 1 TO 5:
PRINT B$:
NEXT IB
4445 RETURN
4450 REM
4455 REM CHANGE/REVIEW ACTIVITY SUBROUTINE
4460 REM
4465 IS = 1
4470 IE = IS + 9
4475 IF (IE < NA) THEN 4485
4480 IE = NA
4485 PRINT:
PRINT
4490 PRINT "REVIEW/CHANGE TIMES."
4495 PRINT
4500 HTAB (5):
PRINT "ACTIVITY";
4505 HTAB (20):
PRINT "ACTIVITY"
4510 HTAB (5):
PRINT "NUMBER";
4515 HTAB (20):
   PRINT "TIME"
4520 PRINT
4525 FOR I = IS TO IE
4530 HTAB (5):
   PRINT I;
4535 HTAB (20):
   PRINT T(I)
4540 NEXT I
4545 PRINT
4550 PRINT "WHICH ACTIVITY TIME DO YOU WANT"
4555 INPUT "TO CHANGE? ENTER 0 FOR NONE ";CN
4560 IF CN = 0 THEN 4600
4565 IF (CN > = IS) AND (CN < = IE) THEN 4585
4570 PRINT:
   PRINT "INVALID ENTRY"
4575 FOR ID = 1 TO 1000:
   NEXT ID
4580 GOTO 4485
4585 PRINT "ENTER NEW TIME FOR ACTIVITY ";CN
4590 INPUT T(CN)
4595 GOTO 4485
4600 IF (IE = NA) THEN 4615
4605 IS = IE + 1
4610 GOTO 4470
4615 IS = 1
4620 IE = IS + 9
4625 IF IE < = C THEN 4635
4630 IE = C
4635 PRINT
4640 PRINT "REVIEW/CHANGE PRECEDENCE ASSIGNMENTS"
4645 PRINT
4650 HTAB (5):
   PRINT "NUMBER"
4655 HTAB (15):
   PRINT "A"
4660 HTAB (20):
   PRINT "PRECEDES"
4665 HTAB (35):
   PRINT "B"
4670 PRINT
4675 FOR I = IS TO IE
4680 HTAB (5):
   PRINT I;
4685 HTAB (15):
   PRINT AM(I,1);
4690 HTAB (20):
   PRINT ">>";
4695 HTAB (35):
   PRINT AM(I,2)
4700 NEXT I
4705 PRINT
4710 PRINT "WHICH ACTIVITY DO YOU WANT TO"
4715 INPUT "CHANGE?(ENTER 0 FOR NONE) "; CN
4720 IF CN = 0 THEN 4785
4725 IF (CN >= IS) AND (CN <= IE) THEN 4765
4730 PRINT
4735 PRINT "INVALID ENTRY"
4740 PRINT "ENTRY MUST BE BETWEEN"
4745 PRINT IS;
4750 PRINT " AND ";
4755 PRINT IE
4760 FOR ID = 1 TO 1000:
   NEXT ID:
   GOTO 4635
4765 PRINT "ENTER NEW PRECEDENCE REQUIREMENT."
4770 PRINT "ENTER A,B FOR A PRECEDING B."
4775 INPUT "A,B ";AM(CN,1),AM(CN,2)
4780 GOTO 4635
4785 IF IE = C THEN 4800
4790 IS = IE + 1
4795 GOTO 4620
4800 PRINT :
   PRINT "END OF MATRIX"
4805 PRINT
4810 FOR ID = 1 TO 1000:
   NEXT ID
4815 RETURN
4820 REM
4825 REM HOFFMANN BALANCE MENU SUB
4830 REM
4835 HOME
4840 PRINT "HOFFMANN BALANCING PROCEDURE"
4845 PRINT
4850 PRINT "PROGRAM OPTIONS"
4855 PRINT
4860 PRINT TAB(5)"1. PERFORM FORWARD HOFFMANN BALANCE."
4865 PRINT TAB(5)"2. PERFORM REVERSE HOFFMANN BALANCE."
4870 PRINT
4875 INPUT "ENTER OPTION ";OP
4880 IF (OP = 1) OR (OP = 2) THEN 4910
4885 PRINT B$
4890 PRINT "INVALID ENTRY"
4895 PRINT "ENTER 1 OR 2"
4900 FOR ID = 1 TO 1000:
   NEXT ID
4905 GOTO 4835
4910 ON OP GOSUB 2625,2745
4915 RETURN
4920 REM
4925 REM RANK POSITION WEIGHT MENU SUB
4930 REM
4935 HOME
4940 PRINT "RANK POSITION WEIGHT(RPW)"
4945 PRINT "BALANCING PROCEDURE"
4950 PRINT
4955 PRINT "PROGRAM OPTIONS"
4960 PRINT
4965 PRINT TAB( 5)"1. PERFORM FORWARD RPW BALANCE."
4970 PRINT TAB( 5)"2. PERFORM REVERSE RPW BALANCE."
4975 PRINT
4980 INPUT "ENTER OPTION ";OP
4985 IF (OP = 1) OR (OP = 2) THEN 5015
4990 PRINT B$
4995 PRINT "INVALID ENTRY"
5000 PRINT "ENTER 1 OR 2"
5005 FOR ID = 1 TO 1000:
      NEXT ID
5010 GOTO 4935
5015 ON OP GOSUB 5625,5735
5020 RETURN
5025 REM
5030 REM AUTO TEST MENU SUB
5035 REM
5040 HOME
5045 PRINT "AUTOMATIC TEST OPTIONS"
5050 PRINT
5055 PRINT TAB( 5)"1. GENERATE RPW BALANCES."
5060 PRINT TAB( 5)"2. GENERATE HOFFMANN BALANCES."
5065 PRINT TAB( 5)"3. BOTH 1 AND 2."
5070 PRINT TAB( 5)"4. GENERATE MOD HOFFMANN BALANCES."
5075 PRINT
5080 INPUT "ENTER OPTION ";OP
5085 IF (OP > = 1) AND (OP < = 4) THEN 5115
5090 PRINT B$
5095 PRINT "INVALID ENTRY"
5100 PRINT "ENTER AN OPTION 1-4"
5105 FOR ID = 1 TO 1000:
      NEXT ID
5110 GOTO 5040
5115 ON OP GOSUB 5125,4235,5320,6245
5120 RETURN
5125 REM
5130 REM RPW TEST GENERATOR
5135 REM
5140 AU = 1
5145 PRINT "ENTER THE NUMBER OF DIFFERENT"
5150 INPUT "CYCLE TIMES THAT WILL BE USED ";CC
5155 IF (CC > 0) AND (CC < 10) THEN 5190
5160 PRINT B$
5165 PRINT "INVALID ENTRY"
5170 PRINT " THE NUMBER OF CYCLE TIMES MUST BE"
5175 PRINT "BETWEEN 1 AND 10."
5180 FOR ID = 1 TO 1000:
5185 NEXT I
5190 FOR I = 1 TO CC
5195 PRINT "ENTER CYCLE TIME ";I;
5200 INPUT CT(I)
5205 NEXT I
5210 PRINT "DO YOU WANT TO PRINT STATION"
5215 INPUT "ASSIGNMENTS? ENTER Y OR N ";YN$
5220 PA = 1
5225 IF YN$ = "N" THEN PA = 0
5230 GOSUB 3820
5235 PC = 0
5240 FOR AI = 1 TO CC
5245 CT = CT(AI)
5250 GOSUB 5625
5255 GOSUB 3725
5260 PC = 1
5265 NEXT AI
5270 PC = 0
5275 FOR AI = 1 TO CC
5280 CT = CT(AI)
5285 GOSUB 5735
5290 GOSUB 3725
5295 PC = 1
5300 NEXT AI
5305 AU = 0:
PC = 0:
PA = 1
5310 FOR IB = 1 TO 5:
   PRINT B$:
   NEXT IB
5315 RETURN
5320 REM
5325 REM SUBROUTINE TO GENERATE
5330 REM RPW BALANCES AND HOFFMANN
5335 REM BALANCES FOR DELAY FACTORS
5340 REM OF 0.0 TO 2.0 BY 0.5
5345 REM
5350 REM CAN BE REPEATED FOR UP TO
5355 REM 10 CYCLE TIMES
5360 REM
5365 AU = 1
5370 PRINT "ENTER THE NUMBER OF DIFFERENT"
5375 PRINT "CYCLE TIMES THAT WILL BE USED"
5380 INPUT "MAXIMUM OF 10 "; CC
5385 IF (CC > 0) AND (CC <= 10) THEN 5420
5390 PRINT B$
5395 PRINT "INVALID ENTRY"
5400 PRINT "THE NUMBER OF CYCLE TIMES MUST"
5405 PRINT "BE BETWEEN 1 AND 10."
5410 FOR ID = 1 TO 1000:
5415 GOTO 5370
5420 FOR I = 1 TO CC
5425 PRINT "ENTER CYCLE TIME ";I;
5430 INPUT CT(I)
5435 NEXT I
5440 PRINT "DO YOU WANT TO PRINT STATION"
5445 INPUT "ASSIGNMENTS? ENTER Y OR N "; YN$
5450 PA = 1
5455 IF YN$ = "N" THEN PA = 0
5460 GOSUB 3820
5465 PC = 0
5470 FOR AI = 1 TO CC
5475 CT = CT(AI)
5480 GOSUB 5625
5485 GOSUB 3725
5490 NS = 1:
5495 GOSUB 2625
5500 IF DF < > 0 THEN 5510
5505 NS = SN
5510 GOSUB 3725
5515 IF DF = 2 THEN 5530
5520 DF = DF + 0.5
5525 GOTO 5495
5530 NEXT AI
5535 PC = 0
5540 FOR AI = 1 TO CC
5545 CT = CT(AI)
5550 GOSUB 5735
5555 GOSUB 3725
5560 NS = 1:
   DF = 0:
   PC = 1
5565 GOSUB 2745
5570 IF DF <> 0 THEN 5580
5575 NS = SN
5580 GOSUB 3725
5585 IF DF = 2 THEN 5600
5590 DF = DF + 0.5
5595 GOTO 5565
5600 NEXT AI
5605 AU = 0:
5610 PC = 0
5615 FOR IB = 1 TO 5:
   PRINT B$:
NEXT IB
5620 RETURN
5625 REM
5630 REM FORWARD RPW BALANCE SUB
5635 REM
5640 PRINT
5645 IF AU = 1 THEN 5655
5650 INPUT "ENTER CYCLE TIME "; CT
5655 EP  = NA
5660 SP  = 1
5665 BT  = 1:
5670 BM  = 1
5675 AP  = 1
56775 AF = 2
5680 PRINT "CYCLE TIME "; CT
5685 PRINT :
   PRINT :
   PRINT
5690 PRINT "PERFORMING FORWARD RPW BALANCE"
5695 PRINT :
   PRINT :
   PRINT
5700 IF PC = 1 THEN 5715
5705 GOSUB 2965:
   REM DETERMINE IPN
5710 GOSUB 5875:
   REM DETERMINE RPW'S
5715 GOSUB 5940
5720 PRINT "BALANCE COMPLETE"
5725 FOR I = 1 TO 1000:
NEXT I
5730 RETURN
5735 REM
5740 REM REVERSE RPW BALANCE SUBROUTINE
5745 REM
5750 PRINT
5755 IF AU = 1 THEN 5765
5760 INPUT "ENTER CYCLE TIME "; CT
5765 SP = NA
5770 EP = 1
5775 BT = -1:
BM = 1
5780 AP = 2
5785 AF = 1
5790 PRINT:
PRINT:
PRINT
5795 PRINT "CYCLE TIME "; CT
5800 PRINT "PERFORMING REVERSE RPW BALANCE"
5805 PRINT:
PRINT:
PRINT
5810 IF PC = 1 THEN 5825
5815 GOSUB 2965
5820 GOSUB 5875
5825 GOSUB 5940
5830 FOR I = 1 TO SN:
RT(I) = TL(SN - I + 1):
NEXT I
5835 FOR I = 1 TO SN:
TL(I) = RT(I):
NEXT I
5840 REM REVERSING SN ORDER
5845 FOR I = 1 TO NA
5850 AA(I) = SN - AA(I) + 1
5855 NEXT I
5860 PRINT "BALANCE COMPLETE"
5865 FOR ID = 1 TO 1000:
NEXT ID
5870 RETURN
5875 REM
5880 REM DETERMINE RANK POSITION WEIGHT SUBROUTINE
5885 REM
5890 FOR L = 1 TO NA
5895 RPW(L) = T(L)
5900 NEXT L
5905 FOR K = EP TO SP STEP (O - BT)
5910 FOR L = 1 TO C
5915 IF AM(L, AF) < > K THEN 5925
5920 RPW(AM(L, AP)) = RPW(AM(L, AP)) + RPW(K)
5925 NEXT L
5930 NEXT K
5935 RETURN
5940 REM
5945 REM PERFORM RPW BALANCE SUB
5950 REM
5955 FOR I = 1 TO NA
5960 CPN(I) = IPN(I)
5965 NEXT I
5970 SN = 1:
5975 REM INITIALIZE VARIABLES
5980 TL(SN) = CT
5985 CRPW = 0
5990 AS = 0
5995 PRINT "PERFORMING BALANCE STATION "; SN
6000 REM SELECT TASK WITH HIGHEST
6005 REM RPW THAT MEETS THE TIME
6010 REM AND PRECEDENCE RESTRICTIONS
6015 FOR J = SP TO EP STEP BT
6020 IF CPN(J) < > 0 THEN 6050
6025 TRPW = RPW(J)
6030 IF TRPW < = CRPW THEN 6050
6035 CRPW = TRPW
6040 AS = 1
6045 SEL = J
6050 NEXT J
6055 IF AS < > 0 THEN 6075
6060 SN = SN + 1:
6065 REM STATION FULL START NEW STATION
6070 REM DECREASE TIME LEFT BY TASK ASSIGNED
6075 TL(SN) = TL(SN) - T(SEL)
6080 AA(SEL) = SN:
6085 REM ASSIGN TASK TO STATION
6090 REM ESTABLISH NEW PRECEDENCE
6095 REM NUMBERS BASED ON TASK SELECTION
6100 FOR K = 1 TO C
6105 IF AM(K, AP) < > SEL THEN 6115
6110 CPN(AM(K, AF)) = CPN(AM(K, AF)) - 1
6115 NEXT K
6120 CPN(SEL) = - 10
6125 REM TEST FOR ALL TASK ASSIGNED
REM IE FOR ALL CPN'S = -10
FOR I = 1 TO NA
IF CPN(I) = -10 THEN 6150
GOTO 5980:
REM SELECT ANOTHER TASK
NEXT I
REM ALL TASK ASSIGNED
REM BALANCE COMPLETE
REM DETERMINE STANDARD DEVIATION
REM OF SLACK TIMES-TL(I)
SU = 0
SS = 0
FOR I = 1 TO SN
SU = SU + TL(I)
SS = SS + (TL(I) * TL(I))
NEXT I
SD = ((SS - ((SU * SU) / SN) / SN) ^ 0.5
REM DETERMINE LINE EFFICIENCY
ST = 0
FOR I = 1 TO NA
ST = ST + T(I)
NEXT I
EFF = (ST / (CT * SN)) * 100
RETURN
REM MOD HOFFMANN BALANCE GENERATOR
REM SUBROUTINE. THE INITIAL NUMBER
REM OF STATIONS IS DETERMINED BY
REM THE NEXT WHOLE NUMBER FROM THE
REM RATIO OF THE SUM OF TASK TIMES.
REM TO CYCLE TIMES.
REM MOD HOFFMANN BALANCE GENERATOR
REM SUBROUTINE.
PRINT "MOD HOFFMANN"
PRINT "BALANCE GENERATOR"
PRINT
PRINT "ENTER THE NUMBER OF DIFFERENT"
PRINT "CYCLE TIMES THAT WILL BE USED."
INPUT "MAXIMUM OF 10 "; CC
IF (CC > 0) AND (CC = 10) THEN 6355
PRINT B$ 6325 PRINT B$ 6330 PRINT "INVALID ENTRY"
PRINT "THE NUMBER OF CYCLE TIMES MUST" 6340 PRINT "BE BETWEEN 1 AND 10."
FOR ID = 1 TO 1000:
NEXT ID
GOTO 6285
FOR I = 1 TO CC
PRINT "ENTER CYCLE TIME ";I;
INPUT CT(I)
NEXT I
PRINT
PRINT "ENTER THE NUMBER OF DELAY FACTORS"
PRINT "THAT WILL BE TESTED"
INPUT "MAXIMUM NUMBER OF 10 "; DD
IF (DD > 0) AND (DD <= 10) THEN 6430
PRINT B$
PRINT "INVALID ENTRY"
PRINT "THE NUMBER OF DELAY FACTORS MUST 
BE BETWEEN 1 AND 10"
FOR ID = 1 TO 1000:
NEXT ID
GOTO 6380
FOR I = 1 TO DD
PRINT "ENTER DELAY FACTOR "; I;
INPUT DF(I)
NEXT I
PRINT "DO YOU WANT TO PRINT STATION"
INPUT "ASSIGNMENTS? ENTER Y OR N "; YN$
PA = 1
IF YN$ = "N" THEN PA = 0
GOSUB 3820
PC = 0
AU = 1
ST = 0
FOR I = 1 TO NA:
ST = ST + T(I):
NEXT I
FOR AI = 1 TO CC
CT = CT(AI)
MNP = INT (ST / CT + 1)
IF MNP = (ST / CT + 1) THEN MNP = ST / CT
NS = MNP
FOR AJ = 1 TO DD
DF = DF(AJ)
GOSUB 2625
D$; "PR#1"
PRINT D$; "MOD HOFFMANN BALANCE"
GOSUB 3725
PC = 1
NEXT AJ
PC = 0
FOR AJ = 1 TO DD
6575 DF = DF(AJ)
6580 GOSUB 2745
6585 PRINT D$;"PR#1"
6590 PRINT "MOD HOFFMANN BALANCE"
6595 PRINT D$;"PR#0"
6600 GOSUB 3725
6605 PC = 1
6610 NEXT AJ
6615 PC = 0
6620 NEXT AI
6625 AU = 0:
   PA = 1
6630 FOR IB = 1 TO 5:
   PRINT B$:
   NEXT IB
6635 RETURN
LIST OF REFERENCES


