Microsoftware for Facility Layout

James D. Anderson

University of Central Florida

This Masters Thesis (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of STARS. For more information, please contact STARS@ucf.edu.

STARS Citation
https://stars.library.ucf.edu/rtd/608
MICROSOFTWARE FOR FACILITY LAYOUT

BY

JAMES DEAN ANDERSON
B.S.E., Union College, Schenectady, N.Y., 1980

THESIS
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

Summer Term 1982
ABSTRACT

This study presents five supporting programs in the area of facility layout design. Programs are "stand alone" adapted type which cover data analysis and analytical techniques. Programs are From/To (From/To Chart generator), CRAFT* (Computerized Relative Allocation of Facilities Technique), RELVAL (Relationship Chart generator), CORELAP* (Computerized Relationship Layout Planning), and EVAL (Layout Evaluation program). Theory, complete program documentation, and case example for each program are presented. Programs are written in an interactive basic mode and have been tested on the Apple II Plus system.

* CRAFT, CORELAP are heuristic techniques for plant design.
To my parents,

James W. Anderson and Virginia Anderson
ACKNOWLEDGEMENTS

I would like to extend my appreciation, first of all, to my advisor, Dr. Yasser A. Hosni who offered me invaluable help and guidance. Acknowledgements are also extended to the following individuals for their willingness to serve on my committee: Dr. Jose Sepulveda and Dr. Gary Whitehouse.

I am greatly indebted to "Lee" who has typed this paper quickly, efficiently, and with great patience.

Finally, I must thank a special friend, Angela R. Barker for her sincere encouragement and motivation.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>Microcomputers Vs. Larger Counterparts</td>
<td>1</td>
</tr>
<tr>
<td>Organization of the Study</td>
<td>3</td>
</tr>
<tr>
<td>II. FROM/TO CHART</td>
<td>6</td>
</tr>
<tr>
<td>Introduction</td>
<td>6</td>
</tr>
<tr>
<td>Program Documentation</td>
<td>8</td>
</tr>
<tr>
<td>Program Listing</td>
<td>9</td>
</tr>
<tr>
<td>Case Example</td>
<td>13</td>
</tr>
<tr>
<td>III. CRAFT</td>
<td>21</td>
</tr>
<tr>
<td>Introduction</td>
<td>21</td>
</tr>
<tr>
<td>Advantages of CRAFT</td>
<td>22</td>
</tr>
<tr>
<td>Limitations of CRAFT</td>
<td>22</td>
</tr>
<tr>
<td>Microsoftware Adaptation for CRAFT Pairwise Exchange Technique</td>
<td>22</td>
</tr>
<tr>
<td>Algorithm and Logic</td>
<td>28</td>
</tr>
<tr>
<td>Program Documentation</td>
<td>31</td>
</tr>
<tr>
<td>Program Listing</td>
<td>33</td>
</tr>
<tr>
<td>Case Example</td>
<td>46</td>
</tr>
<tr>
<td>IV. REL CHART</td>
<td>53</td>
</tr>
<tr>
<td>Introduction</td>
<td>53</td>
</tr>
<tr>
<td>Program Documentation</td>
<td>55</td>
</tr>
<tr>
<td>Program Listing</td>
<td>56</td>
</tr>
<tr>
<td>Case Example</td>
<td>62</td>
</tr>
<tr>
<td>V. CORELAP</td>
<td>71</td>
</tr>
<tr>
<td>Introduction</td>
<td>71</td>
</tr>
<tr>
<td>Program Documentation</td>
<td>76</td>
</tr>
<tr>
<td>Program Listing</td>
<td>78</td>
</tr>
<tr>
<td>Case Example</td>
<td>89</td>
</tr>
</tbody>
</table>
TABLE OF CONTENTS-Continued

<table>
<thead>
<tr>
<th>VI. LAYOUT EVALUATION</th>
<th>..................................................</th>
<th>96</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>..................................................................................</td>
<td>96</td>
</tr>
<tr>
<td>Program Documentation</td>
<td>................................................................................</td>
<td>100</td>
</tr>
<tr>
<td>Program Listing</td>
<td>..................................................................................</td>
<td>102</td>
</tr>
<tr>
<td>Case Example</td>
<td>..................................................................................</td>
<td>111</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VII. CONCLUSIONS AND RECOMMENDATIONS</th>
<th>...............................................</th>
<th>120</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conclusions</td>
<td>..................................................................................</td>
<td>120</td>
</tr>
<tr>
<td>Recommendations</td>
<td>..................................................................................</td>
<td>121</td>
</tr>
</tbody>
</table>

LIST OF REFERENCES............................... 122
CHAPTER I
INTRODUCTION

Facility layout is an area of considerable interest to most industries. Due to modern technology and the increasing number of data variables and constraints involved in the designs, better approaches of developing layout designs are warranted. The development of heuristic analytical approaches has proved effective in producing better designs. In conjunction with these complex techniques, computers in general and microcomputers in particular are being used in this paper in solving layout design problems. A number of microcomputer programs are documented with case example to demonstrate their use in the area of layout arrangement.

Microcomputers Vs. Larger Counterparts

With the advance of technology, the processing power, reliability, size, and speed of the microcomputer has improved remarkably. Furthermore, the cost of a microcomputer system has dropped significantly [9].

Although microcomputers and large computers are similar in many ways, microcomputers do have characteristics that distinguish them from large general-purpose computers. Microcomputers are small, inexpensive, and personal. Presently, microcomputers are not "yet"
recommended for applications requiring large data base or high Input/Output rates, which characterize most large commercial processing jobs. However, the capability of microcomputers has increased by the addition of more memory, peripherals, and software so that some configurations offer almost all the capabilities of larger systems for cases of "stand alone" programs which characterize many industrial engineering techniques [1], [5].

Practicing Industrial Engineers will find it beneficial to obtain an inexpensive microcomputer to solve medium to small size problems within their department, such as facility layout and production departments arrangement. Rapid changes in industry, product variations, product volume, and other factors require continuous analysis and evaluation of existing layout with respect to meeting the requirements of such changes. This information, along with its cost, is required by management in the decision process pertinent to production systems. This information should be readily available for cases such as accepting or rejecting offers of producing new products, or shop scheduling in production plants. Use of microcomputers could result in quicker, cheaper, and more effective analysis of layout changes if the supporting software is available [1].

This study presents five supporting programs in the area of facility layout, that will aid the designer using a microcomputer to analyze his/her data and apply known analytical techniques resulting in an effective layout.
Existing computer assisted layout techniques can be classified into two main categories: Construction type algorithm, in which the arrangement of departments starts without an initial layout and uses the program logic until a layout design is achieved. The second category referred to as the improvement-type algorithm, requires an initial layout as input and the rearrangement of the different departments of the layout occurs in order to obtain an improved layout design.

Organization of the Study

The next five chapters document 5 programs used in the facility layout arrangement. Each program is documented through the theory behind it, function of the program, input requirements, logic, code, and the output. Use of the programs are demonstrated through an example at the end of each chapter.

Chapter II presents a data analysis program. From/To Chart is a program which provides information concerning the number of handling trips made between each department center of activity. This analysis is based on different sequences of operations for a number of products with varying quantities and handling batch sizes.

Chapter III presents a layout improvement program. CRAFT (Computerized Relative Allocation of Facilities Technique) seeks an optimum design, with respect to cost of material handling, by making improvements in an existing layout. CRAFT uses a pairwise exchange routing as its basic technique.
Although Chapters II & III are documenting two separate programs, the user can use them as one unit, i.e., the From/To Chart produces the information needed as input for CRAFT.

Chapter IV presents a data analysis program. REL Chart is a management tool designed to facilitate considerations of qualitative factors in their designs. The REL Chart replaces the quantitative data in a From/To Chart by qualitative closeness ratings.

Chapter V presents a construction program. CORELAP (Computerized Relationship Layout Planning) considers the qualitative factors influencing the arrangement of departments as represented in the REL Chart, and produces a closeness rating value for each department which is used in the departmental arrangement.

Although Chapters IV & V are documenting two separate programs, the user can use them as one unit, i.e., the REL Chart produces the information needed as input for CORELAP.

Chapter VI presents a layout evaluation program and provides guidelines for rearrangement with respect to satisfying closeness ratings. Programs are designed in a way such that no programming experience is required by users. Instructions are given on how to use the program, input data, and select program options. Computer memory space for data storage is allocated according to the size of the user's problem and is limited only by the capacity of the microcomputer in use. Programs are written in Applesoft Basic computer language, which makes it directly usable on an Apple II microcomputer system. Minor adjustments are required in the code
to make it usable on other systems.
CHAPTER II
FROM/TO CHART

Introduction
The From/To Chart, also referred to as a travel chart and a cross chart, is an adaptation of the familiar mileage chart appearing on most road maps. Typically, the From/To Chart provides information concerning the number of material handling trips made between two centers of activity. It is used by the industrial engineers in designing and analyzing production plants. Its primary use is in the departmental arrangement and in the computation of material-handling cost [4].

Products are moved between departments in batches. During this process there may be variations in volumes to produce, differences in batch sizes, and the sequence of operation for each part. In cases like this, microcomputers can greatly reduce the computational effort of number of trips made by material handling equipment used and consequently the cost of handling.

The analyst assumes that the material-handling cost is proportional to the number of trips exchanged between departments when using the From/To Chart for departmental arrangement. Because of this proportionality, the departments should be arranged in a way that will increase the proximity between "active" departments [5].
(activity with respect to the number of trips exchanged between departments).

The cost(s) per trip is dependent on the distance traveled. If this cost and a unit distance between different departments are known, then the total cost of material handling for a specific arrangement can be computed.

From/To Charts are used in conjunction with process layouts and have been found to be useful in the following:

1. "Selling" the layout arrangement to the management.
2. Analyzing material movement.
3. Developing departmental block plans.
4. Developing detailed layout arrangements.
5. Evaluating layout alternatives.
6. Demonstrating the dependency of one area upon another.
7. Improving the use of floor space.
8. Showing interrelationship of product lines.
Program Name: From/To Chart.

Program Function: The From/To Chart provides the number of material-handling trips made between each two departments in a production plant.

Program Input: 1. Number of product types to be produced.
2. Number of production departments.
3. The quantity associated for each part/product.
4. The batch size associated for each part/product.
5. The sequence of operations associated for each part/product expressed in department numbers.

Program Logic: 1. Accept input.
2. Display data, make corrections if any, if not, continue.
3. For each product/part, the # of trips are computed.
   
   \[
   \text{# of trips} = \frac{\text{quantity}}{\text{batch size}}.
   \]
4. In a matrix NXN (Where N = No. of departments involved), the # of trips exchanged between each two departments are accumulated, based on scanning the sequence of operations for each product/part.
5. Make modifications in input data or exit program.

Program Output: From/To Chart with respect to the product (Refer to the case example).

* Self explanatory messages are displayed when the wrong data type is used.
Program Listing

10 REM **PLAN: LAYOUT-FROM/TO CART
15 GOSUB 475
20 CLEAR
25 ZZ = 0
30 PRINT : PRINT : PRINT
35 INPUT "ENTER # OF PARTS TO BE PRODUCED ";NP
40 DIM P(NP,50),R(NP),B(NP),TP(NP),N(NP)
45 PRINT
50 INPUT "ENTER # OF PRODUCTION DEPARTMENTS ";ND
55 DIM DD(ND,ND)
60 PRINT
65 HOME
70 PRINT "FOR EACH PART ENTER:";
75 PRINT : PRINT " QUANTITY, BATCH SIZE,SEQUENCE"
80 PRINT "; OF OPERATIONS"
85 PRINT : PRINT "NOTE:"; PRINT
90 PRINT " 1- USE DEPT.#'S FOR OPERATIONS"; PRINT
95 PRINT " 2- USED DEPT.#'S CAN NOT EXCEED "; ND
100 PRINT
105 PRINT " 3- ENTER '0' TO END OPERATIONAL SEQ.";
110 PRINT : PRINT " 4- # OF OPERATIONS CAN NOT EXCEED": PRINT
115 PRINT : PRINT : PRINT : PRINT
120 PRINT : PRINT : INPUT "HIT RETURN WHEN READY "; Y$ 
125 HOME
130 FOR J = 1 TO NP
135 GOSUB 140: GOTO 235
140 N(J) = 0:B(J) = 0
145 PRINT "PART "; J
150 INPUT "ENTER QUANTITY ";Q(J)
155 INPUT "ENTER BATCH SIZE ";R(J)
160 IF B(J) = 0 THEN PRINT "BATCH SIZE CAN NOT BE 0, REENTER "; GOTO 155
165 FOR I = 1 TO 50
PRINT "ENTER DEPARTMENT # FOR OPERATION";'I;
175 INPUT "";'P(J,I)
180 IF P(J,I) > NF THEN PRINT "DEPARTMENT # EXCEEDS THE # OF DEPTS."; PRINT "REENTER"; GOTO 170
185 IF I > 1 THEN I1 = I - 1
190 IF P(J,I) = P(J,I1) THEN PRINT "TRIPS CAN NOT BE INSIDE SAME DEPARTMENT"; PRINT "REENTER"; GOTO 170
195 IF P(J,I) = 0 THEN 215
200 N(J) = N(J) + 1
205 NEXT I
210 PRINT : PRINT
215 PRINT : INPUT "DO YOU WANT TO MAKE CHANGES? (Y OR N)?"; B$; IF B$ = "N" THEN 230
220 GOSUB 520
225 GOTO 215
230 RETURN
235 HOME
240 NEXT J
245 REM ** #TRIPS BETWEEN DEPTS ARE CALCULATED
250 HOME : PRINT : PRINT
255 PRINT "YOUR COMPUTER IS CALCULATING"; PRINT "PLEASE STAND BY"
260 FOR J = 1 TO NF
265 T = Q(J) / B(J); I% = T; R = T - I%
270 IF R = 0 THEN TP(J) = T; GOTO 280
275 TP(J) = I% + 1
280 NEXT J
285 FOR J = 1 TO NF
290 M = N(J) - 1
295 FOR I = 1 TO M
300 I1 = I + 1
305 J1 = P(J,I); J2 = P(J,I1)
310 DD(J1,J2) = DD(J1,J2) + TP(J)
315 NEXT I
320 NEXT J
325 HOME : GOSUB 330: GOTO 425
330 PRINT "FROM/TO CHAR T": PRINT "UNITS: # OF TRIPS"
335 PRINT " "
340 IC = 0
345 FOR I = 1 TO ND
350 PRINT "FROM DEPT. ":I
355 FOR J = 1 TO ND
360 PRINT " TO DEPT. ":J!
" # OF TRIPS=";DD(I,J)
365 IF Z% = 1 THEN 400
370 IC = IC + 1
375 IF IC < 12 THEN 400
380 PRINT ; PRINT "HIT 'RETURN' TO CONTINUE ";: GET Y$
385 IC = 0
390 IF J = ND THEN HOME : GOTO 400
395 HOME : PRINT "FROM DEPT. ":I
400 NEXT J
405 NEXT I
410 IF Z% = 1 THEN 420
415 PRINT ; INPUT "HIT 'RETURN' TO CONTINUE ";: Y$: HOME
420 RETURN
425 HOME
430 PRINT "NEXT STEP TO BE PERFORMED:"
435 PRINT " 1. EXIT PROGRAM"
440 PRINT " 2. MODIFY EXISTING DATA"
445 PRINT " 3. INPUT NEW DATA"
450 PRINT ; PRINT ; PRINT
455 PRINT " ENTER CHOICE # "; GET X
460 ON X GOTO 470,600,15
465 PRINT "CHOICE MUST BE 1, 2, 0 OR 3": GOTO 425
470 HOME : END
475 HOME : FOR X = 1 TO 5; PRINT ; NEXT X
480 PRINT " PLANT DESIGN"
485 PRINT " FROM/TO CENTRAL FLORIDA"
490 PRINT " IEMS DEPARTMENT"
495 PRINT " UNIVERSITY OF CENTRAL FLORIDA"
FOR X = 1 TO 3000: NEXT X: HOME

RETURN

STOP

FOR J = J TO NP: N(J) = 0

PRINT : PRINT : PRINT : PRINT

A$(1) = "QUANTITY "; A$(2) = "BATCH SIZE "; A$(3) = "DEPT.
# OF OPERATIONS"

FOR X = 1 TO 3: PRINT X; ":"

?A$(X): NEXT X

PRINT : PRINT : PRINT "ENTER
# OF ITEM TO BE CHANGED "; PRINT
: INPUT X

IF X = 1 THEN 560

IF X = 2 THEN 565

IF X = 3 THEN 570

PRINT : INPUT "ENTER NEW VAL
UE FOR QUANTITY "; Q(J): RETURN

PRINT : INPUT "ENTER NEW VAL
UE FOR BATCH SIZE "; B(J): RETURN

FOR I = 1 TO 50

PRINT "ENTER NEW DEPT. # OF
OPERATION "; I;

INPUT ": "; F(J, I)

IF F(J, I) = 0 THEN RETURN

N(J) = N(J) + 1: NEXT I

RETURN

GOSUB 605: GOTO 250

HOME : PRINT : PRINT : PRINT

PRINT "ENTER PART# FOR WHICH
DATA": INPUT "TO BE MODIFIE
D (0 TO CONTINUE ) "; J

IF J = 0 THEN 625

GOSUB 140: GOTO 605

RETURN
A company consists of seven production departments (#1 through #7). Products are classified into five groups with each having to pass through a specific operation sequence. Table 2.1 displays the volume of production of each group (PSC), the bulk factor (Batch size or pieces/load (PSC/L)), and the operation sequence.

**TABLE 2.1**

PRODUCT INFORMATION

<table>
<thead>
<tr>
<th>GROUP #</th>
<th>VOLUME (PCS)</th>
<th>BULK FACTOR (PCS/L)</th>
<th>OPERATION SEQUENCE (DEP. #)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>700</td>
<td>7</td>
<td>1, 5, 4, 3, 7</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>1</td>
<td>1, 3, 2, 6, 7</td>
</tr>
<tr>
<td>3</td>
<td>1000</td>
<td>20</td>
<td>1, 5, 7</td>
</tr>
<tr>
<td>4</td>
<td>390</td>
<td>30</td>
<td>1, 2, 3, 4, 5, 6, 7</td>
</tr>
<tr>
<td>5</td>
<td>77</td>
<td>7</td>
<td>1, 5, 2, 3, 7</td>
</tr>
</tbody>
</table>

After loading the program and typing RUN, the screen displays the following

```
RUN

PLANT DESIGN
FROM/TO CHART

IEMS DEPARTMENT
UNIVERSITY OF CENTRAL FLORIDA
```
This case example involves the manufacturing of five product/part types within seven production departments. The program begins with the following questions:

ENTER # OF PRODUCT/PART TYPES
TO BE PRODUCED 5

ENTER # OF PRODUCTION DEPARTMENTS 7

This is followed by a brief statement of program instructions.

FOR EACH PART ENTER:

QUANTITY, BATCH SIZE, SEQUENCE OF OPERATIONS

NOTE:

1- USE DEPT. #'S FOR OPERATIONS
2- USED DEPT. #'S CAN NOT EXCEED 7
3- ENTER '0' TO END OPERATIONAL SEQ.
4- # OF OPERATIONS CAN NOT EXCEED 50 PER PART

HIT RETURN WHEN READY
The required data is now entered.

PART # 1
ENTER QUANTITY 100
ENTER BATCH SIZE 7
ENTER DEPARTMENT # FOR OPERATION 1 : 1
ENTER DEPARTMENT # FOR OPERATION 2 : 5
ENTER DEPARTMENT # FOR OPERATION 3 : 4
ENTER DEPARTMENT # FOR OPERATION 4 : 3
ENTER DEPARTMENT # FOR OPERATION 5 : 7
ENTER DEPARTMENT # FOR OPERATION 6 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? Y

The opportunity to edit the data is given after entering data for each product group. The correction options are displayed:

1. QUANTITY
2. BATCH SIZE
3. DEPT. # OF OPERATIONS

Corrections are made by entering the number of the item to be changed and the corresponding value.

ENTER # OF ITEM TO BE CHANGED
? 1
ENTER NEW VALUE FOR QUANTITY 700
The remaining data is now entered.

PART # 2
ENTER QUANTITY 12
ENTER BATCH SIZE 1
ENTER DEPARTMENT # FOR OPERATION 1 : 1
ENTER DEPARTMENT # FOR OPERATION 2 : 3
ENTER DEPARTMENT # FOR OPERATION 3 : 2
ENTER DEPARTMENT # FOR OPERATION 4 : 6
ENTER DEPARTMENT # FOR OPERATION 5 : 7
ENTER DEPARTMENT # FOR OPERATION 6 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? N

PART # 3
ENTER QUANTITY 1000
ENTER BATCH SIZE 20
ENTER DEPARTMENT # FOR OPERATION 1 : 1
ENTER DEPARTMENT # FOR OPERATION 2 : 5
ENTER DEPARTMENT # FOR OPERATION 3 : 7
ENTER DEPARTMENT # FOR OPERATION 4 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? N

PART # 4
ENTER QUANTITY 390
ENTER BATCH SIZE 30
ENTER DEPARTMENT # FOR OPERATION 1 : 1
ENTER DEPARTMENT # FOR OPERATION 2 : 2
ENTER DEPARTMENT # FOR OPERATION 3 : 3
ENTER DEPARTMENT # FOR OPERATION 4 : 4
ENTER DEPARTMENT # FOR OPERATION 5 : 5
ENTER DEPARTMENT # FOR OPERATION 6 : 6
ENTER DEPARTMENT # FOR OPERATION 7 : 7
ENTER DEPARTMENT # FOR OPERATION 8 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? N

PART # 5
ENTER QUANTITY 77
ENTER BATCH SIZE 7
ENTER DEPARTMENT # FOR OPERATION 1 : 1
ENTER DEPARTMENT # FOR OPERATION 2 : 5
ENTER DEPARTMENT # FOR OPERATION 3 : 2
ENTER DEPARTMENT # FOR OPERATION 4 : 3
ENTER DEPARTMENT # FOR OPERATION 5 : 7
ENTER DEPARTMENT # FOR OPERATION 6 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? N
While the computer is calculating the number of trips between departments, the program will display

**YOUR COMPUTER IS CALCULATING**  
**PLEASE STAND BY**

For the period in which computation is taking place. The results are then displayed to the user.

<table>
<thead>
<tr>
<th>FROM/TO CHART</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNITS: # OF TRIPS</td>
</tr>
<tr>
<td>---------------</td>
</tr>
</tbody>
</table>

| FROM DEPT. 1 | TO DEPT. 1 # OF TRIPS= 0 |
| | TO DEPT. 2 # OF TRIPS= 13 |
| | TO DEPT. 3 # OF TRIPS= 12 |
| | TO DEPT. 4 # OF TRIPS= 0 |
| | TO DEPT. 5 # OF TRIPS= 161 |
| | TO DEPT. 6 # OF TRIPS= 0 |
| | TO DEPT. 7 # OF TRIPS= 0 |

| FROM DEPT. 2 | TO DEPT. 1 # OF TRIPS= 0 |
| | TO DEPT. 2 # OF TRIPS= 0 |
| | TO DEPT. 3 # OF TRIPS= 24 |
| | TO DEPT. 4 # OF TRIPS= 0 |
| | TO DEPT. 5 # OF TRIPS= 0 |
| | TO DEPT. 6 # OF TRIPS= 12 |
| | TO DEPT. 7 # OF TRIPS= 0 |
FROM DEPT. 3
TO DEPT. 1 # OF TRIPS= 0
TO DEPT. 2 # OF TRIPS= 12
TO DEPT. 3 # OF TRIPS= 0
TO DEPT. 4 # OF TRIPS= 13
TO DEPT. 5 # OF TRIPS= 0
TO DEPT. 6 # OF TRIPS= 0
TO DEPT. 7 # OF TRIPS= 111

FROM DEPT. 4
TO DEPT. 1 # OF TRIPS= 0
TO DEPT. 2 # OF TRIPS= 0
TO DEPT. 3 # OF TRIPS= 100
TO DEPT. 4 # OF TRIPS= 0
TO DEPT. 5 # OF TRIPS= 13
TO DEPT. 6 # OF TRIPS= 0
TO DEPT. 7 # OF TRIPS= 0

FROM DEPT. 5
TO DEPT. 1 # OF TRIPS= 0
TO DEPT. 2 # OF TRIPS= 11
TO DEPT. 3 # OF TRIPS= 0
TO DEPT. 4 # OF TRIPS= 100
TO DEPT. 5 # OF TRIPS= 0
TO DEPT. 6 # OF TRIPS= 13
TO DEPT. 7 # OF TRIPS= 50

FROM DEPT. 6
TO DEPT. 1 # OF TRIPS= 0
TO DEPT. 2 # OF TRIPS= 0
TO DEPT. 3 # OF TRIPS= 0
TO DEPT. 4 # OF TRIPS= 0
TO DEPT. 5 # OF TRIPS= 0
TO DEPT. 6 # OF TRIPS= 0
TO DEPT. 7 # OF TRIPS= 25

FROM DEPT. 7
TO DEPT. 1 # OF TRIPS= 0
TO DEPT. 2 # OF TRIPS= 0
TO DEPT. 3 # OF TRIPS= 0
TO DEPT. 4 # OF TRIPS= 0
TO DEPT. 5 # OF TRIPS= 0
TO DEPT. 6 # OF TRIPS= 0
TO DEPT. 7 # OF TRIPS= 0

HIT 'RETURN' TO CONTINUE
Finally, program options are now displayed

NEXT STEP TO BE PERFORMED:
1. EXIT PROGRAM
2. MODIFY EXISTING DATA
3. INPUT NEW DATA

ENTER CHOICE #
<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>13</td>
<td>12</td>
<td>0</td>
<td>161</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>24</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>12</td>
<td>0</td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>111</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>13</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>11</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>13</td>
<td>50</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 2.1. From/To Chart summarizing number of material handling trips per day between departments using case example.
CHAPTER III

CRAFT

Introduction

CRAFT (Computerized Relative Allocation of Facilities Techniques) was originally presented by Armour and Buffa [2]. This program is considered an improvement type as it requires an initial input arrangement. The criterion employed in CRAFT is the minimization of the cost of material handling, where this cost is expressed as a linear function of distance traveled. Input requirements for CRAFT include:

1. Initial layout arrangement showing all departments.
2. Number of trips between the departments.
3. Cost of one trip a unit distance between departments.

CRAFT attempts to make improvements in the initial layout in a sequential fashion. It evaluates the given layout by calculating the cost of material movement, then it considers the effect of interchanging the departments locations. If improvements can be made by making pairwise exchanges, the exchange producing the greatest improvement is made. The process continues until no improvement can be made by pairwise exchanges. This procedure may be characterized as a steepest-descent procedure because it makes that interchange which results in the greatest decrease in the
Advantages of CRAFT

1. The developed layout usually results in an improvement over the initial layout with respect to material-handling cost.
2. CRAFT's evaluation technique proved to be efficient only with respect to quantifiable criteria.

Limitations of CRAFT

1. The algorithm is path-oriented with respect to the initial arrangement (i.e. The result is not necessarily optimal).
2. Craft uses the department centers in measuring the trip distances, which are not realistic for some applications (refer to Figure 3.1).
3. The original CRAFT algorithm used in main frame computers usually produces an output with "zig-zagging" departmental boundaries which might require effort in correcting and readjusting these boundaries to an acceptable arrangement (refer to Figure 3.2).
4. CRAFT deals with quantifiable criteria, neglecting any qualitative factors affecting the design.
5. The improvements from the algorithm are a result of the assumption that the cost of movement has a direct linear relationship with the distance.

Microsoftware Adaptation For CRAFT

Pairwise Exchange Technique

Due to limitations which exist for the microcomputers (memory limitation, speed, etc.), and considering the case of production
Figure 3.1. Rectangular area coordinates key.
Figure 3.2. Location pattern displaying "zig-zagging" boundaries of a layout arrangement.
plants where departments are usually arranged in bays served by common material handling equipment, a new method to input the initial layout arrangement was developed [4]. The method is summarized in the following points:

1. Plant is considered rectangle in shape, with the user supplying its width and length, i.e., total area available.

2. Plant departments are arranged in bays, with the user supplying the bay area and number.

3. Considering the previous points, the user enters his initial arrangements naming the "Northwest Corner" department as the first department number and supplies its area. The user then proceeds by naming the departments following the arrow sequence shown in Figure 3.3 (i.e., for the example in Figure 3.3 the user enters 3,15,2,10,9 as the sequence of departments along with the corresponding area of the departments).

It is the function of the program to translate this information into an initial layout and proceed with the computation. It is believed that by using this method, a good amount of work of eliminating the "zig-zagging" of main frame CRAFT is avoided, as well as accommodation of a program the size of CRAFT by the microcomputer. The program also considers the rectilinear distance in computing the distance between two departments (refer to Figure 3.4).
Figure 3.3. Departmental sequence into plant bays.
Figure 3.4. Rectilinear distance from department #3 to department #4.

\[ RD = |X_{C4} - X_{C3}| + |Y_{C4} - Y_{C3}| \]
Algorithm and Logic

Based on the dimensions of the plant area, number of bays desired, department-sequence, and department areas, the bay arrangement, (statement # 395 in program listing), arranges the departments in the plant bays and computes the coordinates of the centers. The algorithm starts the arrangement in the "Northwest" corner of the rectangular area and positions the first department in the sequence into that position. This process continues until the bay is filled [4]. A department might be located in two bays. This situation occurs when locating a department in a bay where it can only be partially accommodated. The routine continues locating the rest of the department in the next bay, i.e., the departmental areas are honored.

The coordinates of each department in the layout are determined by the XY-coordinates subroutine. The departments are all rectangular, except those departments which occupy more than one bay. The coordinates key is shown in Figure 3.1. The routine calculates the minimum rectilinear distances between the departments centers. If a department occupies more than one bay, the \( X_c \) coordinate is located in the bay where the largest area of the department exists.*

A list of the bays, their contents, and departments coordinates are produced as a result of this routine.

* A relative approximation of the program which is debatable in practical cases.
Total cost of material movement between all departments is then computed and retained. A pairwise exchange routine then exchanges each department location with the other departments and cycles through the coordinates and distance routines, computing the total cost of material handling each time. If there is an improvement, the program implements the exchange. The program considers the arrangement "in hand" as an initial layout, repeats from the beginning the program steps, and reports the location of each department expressed as corner coordinates and the total cost of material handling according to this arrangement. The exchange routine is active until there are no more improvements or all exchanges have taken place.
\[ a = \text{given assignment} \]
\[ D = \text{distance matrix} \]
\[ W = \text{weight matrix} \]
\[ TC(a) = \text{total cost of given assignment} \]
\[ e = \text{maximum of 0, the greatest decrease in } TC(a) \]
\[ DTC = \text{difference in total costs of two arrangements} \]

Figure 3.5. Flow chart of steepest - descent pairwise-interchange procedure.
Program Documentation

Program Name: CRAFT (Computerized Relative Allocation of Facilities Technique).

Program Function: CRAFT is layout improvement program that seeks a sub-optimal department arrangement with respect to cost of material.

Program Input: 1. Number of production departments.
               2. Number of trips between departments.
               3. Cost per trip a unit distance.
               4. Length of available area.
               5. Width of available area.
               6. Number of bays.
               7. For a given sequence, the department number and corresponding area.

Program Logic: 1. Accept input (# of trips between departments and cost per trip a unit distance, length and width of available area, # of bays desired).
               2. Display data, accept corrections if any, if not, continue.
               3. Space available is computed.
                  \[ \text{Space available} = \text{length} \times \text{width of available area} \]
               4. Bay span is computed.
                  \[ \text{Bay span} = \frac{\text{width of available area}}{\# \text{ bays}} \]
               5. For a given sequence, department number and area are entered.
               6. Make corrections, if any, if not, continue.
               7. XY - coordinates are computed (statement # 880 in program listing).
8. Rectilinear distances between each department are computed using the centroids of each department.

9. Total handling cost is computed.

Matrix \( C_{ij} \) = cost to move the total product flow between departments for a unit distance.

Matrix \( D_{ij} \) = distances between departmental centers.

Total cost = \( \Sigma C_{ij} D_{ij} \)

10. Compare this total handling cost with the previous value. If the new value is less than the previous one, implement the exchange and begin exchange subroutine again.

11. Exchange subroutine uses pairwise exchange routine as its basic technique and cycles to step #7. The exchange process continues until no improvement can be made.

12. Final, improved layout arrangement is computed and displayed.

13. XY - coordinates are displayed.

14. Rectilinear distances between each department are displayed.

15. Improved handling cost is computed and displayed.

16. Exit program.

Program Output: Improved layout arrangement, XY - coordinates given as the corner coordinates of each department, distance matrix, and handling cost. (Refer to case example).
Program Listing

10 REM **PLANT LAYOUT-FROM/TO C HART
15 GOSUB 60
20 CLEAR
25 Z% = 0
30 PRINT : PRINT : PRINT
35 PRINT
40 INPUT "ENTER # OF PRODUCTION DEPARTMENTS " ; ND
45 DIM ID(ND:ND),CS(ND:ND),CT(ND:ND)
50 GOTO 105
55 HOME
60 HOME : FOR X = 1 TO 5 : PRINT : NEXT X
65 PRINT " PLANT DESIGN"
70 PRINT " CRAFT"
75 PRINT
80 PRINT " ITEMS DEPARTMENT"
85 PRINT " UNIVERSITY OF CENTRAL FLORIDA"
90 FOR X = 1 TO 3000 : NEXT X : HOME
95 RETURN
100 STOP
105 HOME : PRINT : PRINT : PRINT
110 PRINT "INPUT # TRIPS BETWEEN DEPARTMENTS"
115 PRINT "AND COST PER TRIP A UNIT DISTANCE" : PRINT : PRINT
120 FOR I = 1 TO ND
125 PRINT "FROM DEPT. ":I
130 FOR J = 1 TO ND
135 IF J = I THEN GOTO 190
140 PRINT " TO DEPT. ":J;"#TRIPS,COST = " : : INPUT ID(I,J),CS(I,J)
145 IC = IC + 1
150 IF IC < 12 THEN 190
155 GOSUB 240
160 PRINT : PRINT "HIT 'RETURN' TO CONTINUE" ; GET Y$
165 IC = 0
170 JR = ND - 1
175 IF I = ND AND J = JR GOTO 20
180 IF J = ND THEN HOME: GOTO 190
185 HOME: PRINT "FROM DEPT. "; I
190 NEXT J
195 NEXT I
200 GOSUB 240
205 HOME: PRINT: PRINT
210 FOR I = 1 TO ND
215 FOR J = 1 TO ND
220 CT(I, J) = CS(I, J) * DD(I, J)
225 NEXT J
230 NEXT I
235 GOTO 295
240 INPUT "DO YOU WANT TO MAKE CHANGES (Y OR N) "; Y$
245 IF Y$ = "Y" GOTO 260
250 IF Y$ = "N" GOTO 285
255 GOTO 240
260 HOME: PRINT: PRINT
265 INPUT "CHANGE DATA FROM DEPT "; I1
270 INPUT " 
TO DEPT "; J1
275 INPUT 
"TRIPS,COST 
"; DD(I1, J1), CS(I1, J1)
280 PRINT: PRINT 
"CHANGE IMPLEMENTED "; GOTO 240
285 HOME
290 RETURN
295 REM ** CRAFT **
300 FOR X = 1 TO 4: PRINT: NEXT X
305 INPUT "ENTER LENGTH OF AVAILABLE AREA "; BL: PRINT
310 INPUT "ENTER WIDTH OF AVAILABLE AREA "; BW: PRINT
315 PRINT "***********************
***********************
320 Z = BL * BW: PRINT "TOTAL SPACE AVAILABLE = "; Z; " SQ. UNIT S"
325 PRINT "**************************
***************"; PRINT
330 INPUT "ENTER # OF BAYS DESIRED":NB
335 PRINT
340 PRINT "*******************************************************************
*******************************************************************"
345 BS = BW / NB
350 PRINT "BAY SPAN = ":BS
355 PRINT "*******************************************************************"; PRINT 
            PRINT
360 PRINT "HIT RETURN TO CONTINUE "; GET Y$;
365 DIM BAC(NB),BL(NB,25),B(25,25)
370 DIM X1(NB,25),Y1(NB,25),X2(NB,25),Y2(NB,25),X3(NB,25),Y3(NB,25)
375 DIM DA(50),DB(50),BL(50),DB(50),FG(10),FB(10)
380 IF JP = 1 GOTO 420
385 HOME
386 FOR X = 1 TO 5: PRINT : NEXT X
390 BR = 0
395 PRINT "*******************************************************************
*******************************************************************"
400 PRINT "ARRANGEMENT BEGINS IN NORTHWEST": PRINT "CORNER OF RECTANGULAR AREA."
405 PRINT "FOR THE FOLLOWING SEQUENCE"
410 PRINT "ENTER DEPT. #, AREA"
415 PRINT "*******************************************************************
*******************************************************************"
420 R1 = 0:R2 = 0:I1 = 0:F = 1:S = 0:BB = 0
425 FOR I = 1 TO NB:BAC(I) = 0: NEXT I
430 FOR I = 1 TO NB:DB(I) = 0: NEXT I
435 TT = 0
440 FOR I = 1 TO NB:FG(I) = 0: NEXT I
442 IF AG = 77 GOTO 465
445 IF JP = 0 GOTO 455
450 IF JP = 1 GOTO 465
455 PRINT "SEQUENCE DEPT. #, AREA"
460 PRINT "-------------------"
465 FOR I = 1 TO NB
470 R1 = O : D1 = O
475 FOR J = F TO ND
480 BB = BB + 1
485 IF JP = 1 GOTO 505
486 IF AG = 77 GOTO 505
490 PRINT "" : BB : INPUT "" : IN(J), DA(J)
495 IF JP = 0 GOTO 505
505 TT = TT + DA(J)
510 IF TT > Z GOTO 820
515 S = S + 1
520 DL(J) = DA(J) / BS
525 BAC(I) = BAC(I) + DL(J)
530 IF BAC(I) < BL GOTO 545
535 IF BAC(I) = BL GOTO 585
540 IF BAC(I) > BL GOTO 625
545 R1 = BL - BAC(I)
550 IF R1 < 10 GOTO 570
555 BL(I, S) = DL(J)
560 B(I, S) = DN(J)
565 GOTO 580
570 DL(J) = DL(J) + R1
575 GOTO 585
580 NEXT J
585 BL(I, S) = DL(J)
590 B(I, S) = DN(J)
595 DB(I) = J - F + 1
600 IF FG(I) = 1 THEN DB(I) = DB(I) + 1
605 F = J + 1
610 FB = 0
615 S = 0
620 GOTO 720
625 R1 = BAC(I) - BL
630 IF R1 < 10 GOTO 710
635 R2 = DL(J) - R1
640 BL(I, S) = R2
645 B(I, S) = DN(J)
650 FB(I) = 1
655 DB(I) = J - F + 1
660 IF FG(I) = 1 THEN DB(I) = DB(I) + 1
665 M = I + 1
670 FG(M) = 1
675  D1 = DN(J)
680  BAC(M) = BAC(M) + R1
685  BL(M+1) = R1
690  B(M,1) = D1
695  S = 1
700  F = J + 1
705  GOTO 720
710  DL(J) = DL(J) - R1
715  GOTO 585
720  NEXT I
722  IF JP = 1 GOTO 745
723  IF AG = 77 GOTO 740
725  PRINT : PRINT
730  INPUT "DO YOU WANT TO MAKE CHANGES, YES OR NO? ":A3$
735  IF A3$ = "YES" THEN GOTO 840
736  IF AG = 77 GOTO 420
740  HOME
745  FOR I = 1 TO NB
750  IF BL(I,1) = 0 GOTO 760
755  GOTO 810
760  J1 = DB(I)
765  J1 = J1 - 1
770  FOR J = 1 TO J1
775  J2 = J + 1
780  BL(I,J) = BL(I,J2)
785  B(I,J) = B(I,J2)
790  NEXT J
795  DB(I) = DB(I) - 1
800  M = I - 1
805  DB(M) = DB(M) - 1
810  NEXT I
812  IF JP = 1 GOTO 905
815  GOTO 875
820  PRINT : PRINT : PRINT "*****
825  ******************
826  *****
827  PRINT "THE TOTAL AVAILABLE SQUARE REQUIREMENTS": PRINT "HAVE BEEN EXCEEDED BY ":TT - Z": SQ.UNITS
830  PRINT "*****************
831  *****************": PRINT
835  END
840  HOME : PRINT : PRINT : PRINT : INPUT "ENTER SEQUENCE # OF DEPT. TO BE CHANGED ":A1
845 PRINT: PRINT "ENTER DEPT.#,
AREA TO BE CHANGED ": PRINT

850 INPUT ":;IN(A1),DA(
A1)
855 PRINT: PRINT "** CHANGE IMP
LEMENTED **": PRINT
856 AG = 77
857 GOTO 725
860 HOME: FOR X = 1 TO 8: PRINT:
NEXT X
865 PRINT " EXCHANGE SUBROU
TINE": PRINT: PRINT: PRINT

870 PRINT " PLEASE STAND
BY"
875 HOME: PRINT: PRINT: PRINT:
PRINT
880 PRINT " SUBROUTINE CALC
ULATING"
885 PRINT
890 PRINT " XY-COORDINAT
ES"
895 FOR X = 1 TO 3000: NEXT X
900 HOME
905 FOR I = 1 TO NB
910 IF I = 1 GOTO 935
915 IF I = 3 GOTO 935
920 F = I / 2
925 F = F * 2
930 IF F = I GOTO 985
935 X1(I,1) = 0
940 N1 = DB(I)
945 XC(I,1) = BL(I,1) / 2
950 FOR J = 1 TO N1
955 X2(I,J) = X1(I,J) + BL(I,J)
960 J1 = J + 1
965 IF J = N1 GOTO 975
970 X1(I,J1) = X2(I,J)
975 NEXT J
980 GOTO 1030
985 N1 = DB(I)
990 X2(I,1) = BL
995 FOR J3 = 1 TO N1 
1000 X1(I,J3) = X2(I,J3) - BL(I,J
3)
1005 J2 = J - 1
1010 J4 = J3 + 1
1015 IF J3 = N1 GOTO 1025
1020 X2(I, J4) = X1(I, J3)
1025 NEXT J3
1030 NEXT I
1035 Y1(I, 1) = BW
1040 FOR I = 1 TO NB
1045 N1 = DB(I)
1050 FOR J = 1 TO N1
1055 Y1(I, J) = Y1(I, 1)
1060 Y2(I, J) = Y1(I, J) - BS
1065 NEXT J
1070 IF I = NB GOTO 1085
1075 I2 = I + 1
1080 Y1(I2, 1) = Y2(I, 1)
1085 NEXT I
1090 FOR I = 1 TO NB
1095 J2 = DB(I)
1100 FOR J = 1 TO J2
1105 XC(I, J) = (X1(I, J) + (BL(I, J)) / 2.0))
1110 NEXT J
1115 NEXT I
1120 HALF = 0.5 * BS
1125 YC(I, 1) = Y1(I, 1) - HALF
1130 J6 = DB(I)
1135 FOR J = 1 TO J6
1140 YC(I, J) = YC(I, 1)
1145 NEXT J
1150 FOR I = 2 TO NB
1155 J6 = DB(I)
1160 FOR J = 1 TO J6
1165 I2 = I - 1
1170 YC(I, J) = YC(I2, 1) - BS
1175 NEXT J
1180 NEXT I
1182 IF JP = 1 GOTO 1210
1185 HOME : PRINT ; PRINT
1190 PRINT "" ; XY-COORDINATES
1195 PRINT "" ; CHART"
1200 PRINT "" ; " ; SEQ ; " ; IEP ; " ; X1 ;
1205 ; " ; Y1 ; " ; X2 ; " ; Y2 ; " ; XC ;
1205 ; " ; YC"
1210 BR = 0
1215  FOR I = 1 TO NB
1220  J1 = DB(I)
1225  FOR J = 1 TO J1
1230  BE = BE + 1
1235  Y1(I,J) = ABS(Y1(I,J) - BE)
1240  Y2(I,J) = ABS(Y2(I,J) - BE)
1245  YC(I,J) = ABS(YC(I,J) - BE)
1250  YC(I,J) = YC(I,J)
1252  IF JP = 1 GOTO 1260
1255  PRINT "";BB;"";B(I,J);TAB(9);X1(I,J);TAB(14);Y1(I,J)
1260  NEXT J
1265  NEXT I
1270  FOR I = 1 TO NB
1275  J1 = DB(I)
1280  M = I + 1
1285  IF M > NB AND JP = 1 GOTO 1
370  IF B(I,J1) = B(M,1) THEN GOTO 1305
1295  NEXT I
1297  IF JP = 1 GOTO 1370
1300  GOTO 1360
1305  M = 0
1310  FOR I = 1 TO NB
1315  J1 = DB(I)
1320  M = I + 1
1322  IF M > NB AND JP = 1 GOTO 1
370  IF M > NB GOTO 1360
1325  IF BL(I,J1) > BL(M,1) GOTO 1345
1330  X(I,J1) = X(M,1);YC(I,J1) = YC(M,1)
1335  GOTO 1350
1340  X(M,1) = X(I,J1);YC(M,1) = YC(I,J1)
1345  GOTO 1350
1350  NEXT I
1355  GOTO 1295
1356  IF JP = 1 GOTO 1370
1360  PRINT : PRINT "HIT 'RETURN'
TO CONTINUE ": GET Y
1365 HOME
1370 FOR I = 1 TO ND
1375 FOR J = 1 TO ND
1380 DS(I,J) = 99999
1385 NEXT J
1390 NEXT I
1395 FOR I = 1 TO NB
1400 J2 = IB(I)
1405 FOR J = 1 TO J2
1410 FOR II = 1 TO NB
1415 J3 = IB(I)
1420 FOR J1 = 1 TO J3
1425 J5 = B(I,J)
1430 J4 = B(I1,J1)
1435 D1 = ABS(XC(I,J) - XC(I1,J
1440 D1 = D1 + ABS(YC(I,J) - YC
1445 IF D1 < = DS(J5,J4) THEN D
S(J5,J4) = D1
1450 NEXT J1
1455 NEXT I1
1460 NEXT J
1465 NEXT I
1466 IF JF = 1 GOTO 1460
1470 PRINT : PRINT " DISTANCE
MATRIX"
1475 PRINT " -------------------"
1480 J = 0
1485 FOR F = 1 TO ND
1490 FOR F1 = 1 TO ND
1495 IF DS(F,F1) < DS(F1,F) THEN
DS(F1,F) = DS(F,F1)
1500 IF DS(F1,F) < DS(F,F1) THEN
DS(F,F1) = DS(F1,F)
1505 J = J + 1
1510 A = A + 5
1512 IF JF = 1 GOTO 1519
1515 PRINT DS(F,F1); TAB( A);
1519 IF J = ND AND JF = 1 THEN J
= 0; GOTO 1525
1520 IF J = ND THEN J = 0; PRINT
1525 NEXT F1
1530 A = 0
1535 NEXT F
1540 REM ** TOTAL HANDLING COST
**
1545 IF JP = 1 GOTO 1590
1550 TC = 0
1555 FOR I = 1 TO ND
1560 FOR J = 1 TO ND
1565 CST = DS(I, J) * CT(I, J)
1570 TC = TC + CST
1575 NEXT J
1580 NEXT I
1585 IF JP = 0 GOTO 1650
1590 T1 = 0
1595 FOR I = 1 TO ND
1600 FOR J = 1 TO ND
1605 CST = DS(I, J) * CT(I, J)
1610 T1 = T1 + CST
1615 NEXT J
1620 NEXT I
1630 IF U = 7 THEN RETURN
1635 IF TC <= T1 THEN RETURN
1640 IF T1 < TC THEN TC = T1; V = 0; PP = 77
1645 RETURN
1650 PRINT: PRINT: PRINT
1655 PRINT "-----------------------------
TOTAL HANDLING COST = "; TC
1660 PRINT "-----------------------------
1670 PRINT "HIT 'RETURN' TO CONTINUE ":: GET Y$
1675 HOME
1680 PP = O
1685 E = 1
1690 G = O
1695 Q = ND - 1
1700 FOR G = 1 TO Q
1705 J1 = G + 1
1710 EN = DN(E)
1715 EA = DA(E)
1720 DN(E) = DN(J1)
1725 DA(E) = DA(J1)
1730 DN(J1) = EN
1735 DA(J1) = EA
1736 IF JP = 1 GOTO 1740
1737 FOR I = 1 TO 5: PRINT: NEXT I
1738 PRINT " EXCHANGE SUBRO
UTINE"
1739 PRINT : PRINT : PRINT "
PLEASE STAND BY"
1740 JP = 1
1745 V = V + 1
1750 GOSUB 380
1752 IF JP = 1 GOTO 1760
1755 PRINT : PRINT "HIT 'RETURN'
TO CONTINUE " : GET Y$
1760 IF PP = 77 GOTO 1680
1765 J1 = G + 1
1770 EN = DN(E)
1775 EA = DA(E)
1780 DN(E) = DN(J1)
1785 DA(E) = DA(J1)
1790 DN(J1) = EN
1795 DA(J1) = EA
1800 U = 7
1805 GOSUB 380
1810 U = 0
1815 NEXT G
1820 HOME
1825 FOR I = 1 TO 4: PRINT : NEXT I
1830 PRINT "**************************
****************************
1835 PRINT " FINAL ARRANGEMENT"
1840 PRINT "**************************
****************************
1845 BB = 0
1850 PRINT "SEQUENCE DEPT. #, AR
EA"
1855 PRINT "---------------------
---": PRINT
1860 FOR J = 1 TO ND
1865 BB = BB + 1
1870 PRINT " " : BB : " ":
DN(J), DA(J)
1875 NEXT J
1880 PRINT : PRINT "HIT 'RETURN'
TO CONTINUE " : GET Y$
1885 HOME : FOR I = 1 TO 5: PRINT
: NEXT I
1890 PRINT " XY-COORDINATES C
HART"
1895 PRINT "------------------" : PRINT : PRINT

1900 PRINT "SEQ ";"DEF ";"X1 ";
   ";"Y1 ";"X2 ";"Y2 ";"XC ";"YC"

1905 PRINT "------------------
   "

1910 BB = 0
1915 FOR I = 1 TO NB
1920 J1 = DB(I)
1925 FOR J = 1 TO J1
1930 BB = BB + 1
1935 PRINT ";BB;" ";B(I,J); TAB(9);X1(I,J); TAB(14);Y1(I,J)
   "; TAB(19);X2(I,J); TAB(24)
   ";Y2(I,J); TAB(29);XC(I,J); TAB(34);YC(I,J)
1940 NEXT J
1945 NEXT I
1950 PRINT : PRINT "HIT ‘RETURN’
   TO CONTINUE" ; : GET Y$
1955 HOME : PRINT : PRINT
1960 PRINT "DISTANCE MA
   TRIX"
1965 PRINT "------------------
   "

1970 J = 0
1975 FOR F = 1 TO ND
1980 FOR F1 = 1 TO ND
1985 IF DS(F,F1) < DS(F1,F) THEN
   
   DS(F1,F) = DS(F,F1)
1990 IF DS(F1,F) < DS(F,F1) THEN
   
   DS(F,F1) = DS(F1,F)
1995 J = J + 1
2000 A = A + 5
2005 PRINT DS(F,F1); TAB( A)
2010 IF J = ND THEN J = 0: PRINT

2015 NEXT F1
2020 A = 0
2025 NEXT F
2030 PRINT : PRINT
2035 PRINT "------------------
   "
2040 PRINT "TOTAL HANDLING COST = " ; TC
2045 PRINT "-----------------------------
2050 PRINT : PRINT "HIT 'RETURN' TO CONTINUE" ; GET Y$
2055 HOME : PRINT : PRINT : PRINT : PRINT "NEXT STEP TO BE PERFORMED"
2060 PRINT " 1.EXIT PROGRAM"
2065 PRINT " 2.INPUT NEW DATA"
2070 PRINT : PRINT : PRINT "ENTER CHOICE # " ; GET X
2075 ON X GOTO 2080,10
2080 END
Case Example

To illustrate the approach used by CRAFT to develop layouts, consider the initial layout and the From/To Chart for the flow data given in Figure 3.6 [4]. For simplicity it will be assumed all cost elements are equal to unity in the From/To Chart for the cost data.

Figure 3.6. a) Initial layout, b) flow data, c) distance data.
After loading the program and typing RUN, the screen displays the following:

This case example involves four production departments. The program begins with the following questions:

ENTER # OF PRODUCTION DEPARTMENTS 4

INPUT # TRIPS BETWEEN DEPARTMENTS
AND COST PER TRIP A UNIT DISTANCE

FROM DEPT. 1
TO DEPT. 2 #TRIPS,COST = ?2,1
TO DEPT. 3 #TRIPS,COST = ?4,1
TO DEPT. 4 #TRIPS,COST = ?4,1
FROM DEPT. 2
TO DEPT. 1 #TRIPS,COST = ?1,1
TO DEPT. 3 #TRIPS,COST = ?1,1
TO DEPT. 4 #TRIPS,COST = ?3,1
FROM DEPT. 3
TO DEPT. 1 #TRIPS,COST = ?2,1
TO DEPT. 2 #TRIPS,COST = ?1,1
TO DEPT. 4 #TRIPS,COST = ?2,1
FROM DEPT. 4
TO DEPT. 1 #TRIPS,COST = ?4,1
TO DEPT. 2 #TRIPS,COST = ?1,1
TO DEPT. 3 #TRIPS,COST = ?0,1
DO YOU WANT TO MAKE CHANGES (Y OR N) Y
The opportunity to make corrections is given after entering the required data. Corrections are made with the format From Dept., To Dept., and the corrected # trips, cost.

CHANGE DATA FROM DEPT. 4
    TO DEPT. 1
    # TRIPS, COST 4,1

** CHANGE IMPLEMENTED **
DO YOU WANT TO MAKE CHANGES (Y OR N)
DO YOU WANT TO MAKE CHANGES (Y OR N) N
HIT 'RETURN' TO CONTINUE; GET Y

The required data pertinent to the layout arrangement is now entered.

ENTER LENGTH OF AVAILABLE AREA 80
ENTER WIDTH OF AVAILABLE AREA 40

The space available is computed.

*******************************************************************************
TOTAL SPACE AVAILABLE = 3200 SQ. UNITS
*******************************************************************************

The number of bays desired in the layout arrangement is now entered.

ENTER # OF BAYS DESIRED 2
The bay span is now computed.

**BAY SPAN = 20**

HIT RETURN TO CONTINUE

The department number and area are now entered for a given initial sequence.

**ARRANGEMENT BEGINS IN NORTHWEST CORNER OF RECTANGULAR AREA.**

**FOR THE FOLLOWING SEQUENCE**

**ENTER: DEPT, #, AREA**

**SEQUENCE DEPT, #, AREA**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1,1000</td>
</tr>
<tr>
<td>2</td>
<td>2,600</td>
</tr>
<tr>
<td>3</td>
<td>4,800</td>
</tr>
<tr>
<td>4</td>
<td>3,800</td>
</tr>
</tbody>
</table>

DO YOU WANT TO MAKE CHANGES, YES OR NO?  NO

The opportunity to make corrections is given after entering the required data.
While the computer is calculating the XY-Coordinates, the program will display

**SUBROUTINE CALCULATING**

**XY-COORDINATES**

for the period in which computation is taking place. The results are then displayed to the user.

**XY-COORDINATES CHART**

<table>
<thead>
<tr>
<th>SEQ</th>
<th>DEP</th>
<th>X1</th>
<th>Y1</th>
<th>X2</th>
<th>Y2</th>
<th>XC</th>
<th>YC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>50</td>
<td>20</td>
<td>25</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>50</td>
<td>0</td>
<td>50</td>
<td>20</td>
<td>65</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>40</td>
<td>20</td>
<td>40</td>
<td>40</td>
<td>60</td>
<td>30</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0</td>
<td>20</td>
<td>40</td>
<td>40</td>
<td>20</td>
<td>30</td>
</tr>
</tbody>
</table>

HIT ‘RETURN’ TO CONTINUE

The rectilinear distances between each department and the handling cost are now computed and displayed.

**DISTANCE MATRIX**

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>40</th>
<th>25</th>
<th>55</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>0</td>
<td>65</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>65</td>
<td>0</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>25</td>
<td>40</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**TOTAL HANDLING COST = 1020**

HIT ‘RETURN’ TO CONTINUE
While the computer is performing pairwise exchange, the program will display

**EXCHANGE SUBROUTINE**

**PLEASE STAND BY**

for the period in which exchanges are being made. The final layout arrangement is then displayed to the user (sequence, areas, coordinates for the 4 corners of each department and rectilinear distance).

```
<table>
<thead>
<tr>
<th>1</th>
<th>4</th>
<th>800</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>600</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>800</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1000</td>
</tr>
</tbody>
</table>
```

HIT 'RETURN' TO CONTINUE
XY-COORDINATES CHART

<table>
<thead>
<tr>
<th>SEQ</th>
<th>DEF</th>
<th>X1</th>
<th>Y1</th>
<th>X2</th>
<th>Y2</th>
<th>XD</th>
<th>YC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>40</td>
<td>20</td>
<td>20</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>40</td>
<td>0</td>
<td>70</td>
<td>20</td>
<td>50</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>70</td>
<td>0</td>
<td>80</td>
<td>20</td>
<td>65</td>
<td>30</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>50</td>
<td>20</td>
<td>80</td>
<td>45</td>
<td>65</td>
<td>30</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
<td>20</td>
<td>50</td>
<td>40</td>
<td>25</td>
<td>30</td>
</tr>
</tbody>
</table>

HIT 'RETURN' TO CONTINUE

DISTANCE MATRIX

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>50</th>
<th>40</th>
<th>25</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0</td>
<td>35</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>30</td>
<td>0</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>25</td>
<td>65</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

TOTAL HANDLING COST = 120

HIT 'RETURN' TO CONTINUE

Finally, program options are now displayed.

NEXT STEP TO BE PERFORMED
1. EXIT PROGRAM
2. INPUT NEW DATA

ENTER CHOICE #
CHAPTER IV

REL CHART

Introduction

Flow analysis tends to relate various activities on a quantitative basis. The relationship is normally expressed as some function of material-handling cost. However, a number of factors other than material-handling cost might be of primary concern in layout design. The activity relationship chart, also referred to as the REL chart was designed to facilitate a consideration of qualitative factors. The REL chart, as developed by Muther [7], replaces the quantitative data in a From/To chart by qualitative closeness ratings.

It is the purpose of this section to present a computer program written in Interactive Basic, which builds a REL Chart equivalent to quantitative material flow expressed in number of trips (Chapter II). The produced REL Chart reflects production requirements expressed in closeness ratings as follows:

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Closeness Rating* and their program numerical values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolutely necessary</td>
<td>A 6</td>
</tr>
<tr>
<td>Especially important</td>
<td>E 5</td>
</tr>
<tr>
<td>Important</td>
<td>I 4</td>
</tr>
<tr>
<td>Ordinary closeness</td>
<td>O 3</td>
</tr>
<tr>
<td>Unimportant</td>
<td>U 2</td>
</tr>
<tr>
<td>Undesireable</td>
<td>X 1</td>
</tr>
</tbody>
</table>
The user must supply the extreme values for the "Absolutely Necessary" relationship and the "Undesireable" relationship if pertinent*.

The creation of the product equivalent relationship chart between the production departments is generated by using product data. Based on each product's sequence of operations, the moving batch size, and the quantities to be produced, a From/To Chart is constructed (refer to Chapter II). The highest value and the lowest value of the number of trips in the From/To Chart are determined. The difference between highest and lowest value is divided into four spans, and a highest value and lowest value of each category are determined. These categories correspond to the A, E, I, O, and U closeness ratings. Each entry within the From/To table is evaluated, and the corresponding closeness rating is assigned. Closeness ratings A and X are to be assigned by the user depending on product design and processes.

If data concerns the cost per trip between various departments, the user might include the effect of varying costs by adjusting the "quantity" value accepted as part of the input data required.

* Ratings are commonly used by layout designer.
Program Name: Product REL Chart.

Program Function: The Product REL Chart provides a representation of a measure of activity relationships by performing pairwise evaluations of the importance for two activities to be located together in the layout.

Program Input: 1. Number of parts to be produced.
2. Number of production departments.
3. The quantity associated for each part.
4. The batch size associated for each part.
5. The sequence of operations associated for each part.

Program Logic: 1. Accept input.

2. Display data, make corrections if any, if not, continue.

3. Relationship chart with respect to the product is formed.
   - From/To Chart is constructed (Refer to Chapter II).
   - Four spans are computed, corresponding to the E, I, O, U Closeness ratings*.
     \[ \text{Span} = \frac{\text{Highest \# trips} - \text{Lowest \# trips}}{4} \]
   - A highest value and lowest value for each category are determined.
   - Each entry within the From/To Chart is evaluated and assigned the corresponding closeness rating.

Program Output: Relationship Chart with respect to the product.

* Closeness ratings A and X are to be assigned by the user.
10 REM **PLANT LAYOUT-DEPARTMENTAL RELATIONSHIP CHART
15 GOSUB 645
20 CLEAR
25 PRINT : PRINT : PRINT
30 INPUT "ENTER # OF PARTS TO BE PRODUCED ": NP
35 PRINT
40 DIM F(NP,50),Q(NP),B(NP),TP(NP),N(NP)
45 INPUT "ENTER # OF PRODUCTION DEPARTMENTS ":ND
50 DIM DD(ND,ND)
55 PRINT
60 HOME
65 PRINT "FOR EACH PART ENTER:": PRINT " QUANTITY, BATCH SIZE, SEQUENCE"
70 PRINT " OF OPERATIONS"
75 PRINT "NOTE:": PRINT
80 PRINT " 1- USE DEPT.#'S FOR OPERATIONS": PRINT
85 PRINT " 2- USED DEPT.#'S CAN NOT EXCEED ":ND
90 PRINT
95 PRINT
100 PRINT " 3- ENTER '0' TO END OPERATIONAL SEQ.": PRINT"ND OPERATIONAL SEQ."
105 PRINT : PRINT " 4- # OF OPERATIONS CAN NOT EXCEED": PRINT " 50 PER PART ": PRINT
110 PRINT : PRINT : PRINT : PRINT
115 PRINT : PRINT : INPUT "HIT RETURN WHEN READY ":Y$ 120 HOME
125 FOR J = 1 TO NP
130 GOSUB 135: GOTO 225
135 N(J) = 0
140 PRINT "PART ": J
145 INPUT "ENTER QUANTITY ":Q(J)
150 INPUT "ENTER BATCH SIZE ":B(J)
155 IF B(J) = 0 THEN PRINT "BATCH SIZE CAN NOT BE 0, REENTER": GOTO 150
160 FOR I = 1 TO 50
165 PRINT "ENTER DEPT.# FOR OPERATION ";I;
170 INPUT ";:F(J,I)
175 IF F(J,I) > ND THEN PRINT "DEPARTMENT # EXCEEDS THE # OF DEPTS."; PRINT "REENTER"; GOTO 165
180 IF I > 1 THEN I1 = I - 1; IF F(J,I) = F(J,I1) THEN PRINT "TRIPS CANNOT BE INSIDE SAME DEPARTMENT"; PRINT "REENTER "; GOTO 165
185 IF F(J,I) = 0 THEN 205
190 N(J) = N(J) + 1
195 NEXT I
200 PRINT ; PRINT
205 PRINT ; INPUT "DO YOU WANT TO MAKE CHANGES, (Y OR N)?";B 
$; IF B$ = "N" THEN 220
210 GOSUB 695
215 GOTO 205
220 RETURN
225 HOME
230 NEXT J
235 REM ** #TRIPS BETWEEN DEPTS ARE CALCULATED
240 PRINT ; PRINT ; PRINT
245 PRINT "YOUR COMPUTER IS CALCULATING"; PRINT "PLEASE STAND BY"
250 FOR J = 1 TO NP
255 T = Q(J) / B(J);I2 = T;R = T - I2
260 IF R = 0 THEN TP(J) = T; GOTO 270
265 TP(J) = I2 + 1
270 NEXT J
275 FOR J = 1 TO NP
280 M = N(J) - 1
285 FOR I = 1 TO M
290 I1 = I + 1
295 J1 = F(J,I);J2 = F(J,I1)
300 DD(J1,J2) = DD(J1,J2) + TP(J)
305 NEXT I
310 NEXT J
315 FOR I = 1 TO ND
320 FOR J = 1 TO ND
325 DD(I,J) = DD(I,J) + DD(J,I)
330 DD(J,I) = DD(I,J)
335 NEXT J
340 NEXT I
345 REM **GENERATE REL CHART
350 MIN = 9999: MAX = 0
355 FOR I = 1 TO ND
360 FOR J = 1 TO ND
365 IF DD(I,J) > MAX THEN MAX = DD(I,J)
370 IF (DD(I,J) < MIN) THEN MIN = DD(I,J)
375 NEXT J
380 NEXT I
385 S = (MAX - MIN) / 4
390 IC = 0
395 FOR I = 1 TO ND
400 FOR J = 1 TO ND
405 FOR I1 = 1 TO 4
410 IF DD(I,J) <= (I1 * S - MIN) THEN DD(I,J) = I1 + 1: GOTO 420
415 NEXT I1
420 NEXT J
425 NEXT I
430 FOR I = 1 TO ND
435 FOR J = 1 TO ND
440 IF I = J THEN DD(I,J) = 0: GOTO 455
445 IF DD(I,J) > DD(J,I) THEN DD(J,I) = DD(I,J): GOTO 455
450 IF DD(J,I) > DD(I,J) THEN DD(I,J) = DD(J,I)
455 NEXT J
460 NEXT I
465 HOME
470 FOR RT = 1 TO 3: PRINT : NEXT RT
475 PRINT "NOTE:": PRINT : PRINT 
"-THE USER MAY SUPPLY CERTAIN RELV AL S": PRINT 
"WHICH WILL AUTOMATICALLY OVERRIDE" 
"THE GENERATED VALUES": PRINT : PRINT
480 PRINT "-RELVAL NO.1 AND NO. 6 MUST BE SUPPLIED": PRINT 
"BY THE USER IF PERTINENT": PRINT 
485 PRINT "DO YOU WANT TO SUPPLY ANY RELV AL S ?": PRINT : INPUT 
"(Y OR N) ":;Y$
590 IF Y$ = "Y" THEN GOSUB 815: GOTO 505
595 IF Y$ = "N" THEN 505
600 GOTO 485
605 HOME: GOSUB 510: GOTO 595
610 PRINT " DEPARTMENTAL RELATIONSHIP CHART": PRINT "---
": PRINT
615 FOR I = 1 TO ND
620 PRINT "BETWEEN DEPT. ":I
625 FOR J = 1 TO ND
630 PRINT " AND DEPT. ":;ID(I,J)
635 IC = IC + 1
640 IF IC < 9 THEN 570
645 GOSUB 775
650 PRINT: PRINT "HIT 'RETURN' TO CONTINUE ": GET Y$
655 IC = 0
660 IF J = ND THEN HOME: GOTO 570
665 HOME: PRINT "BETWEEN DEPT. ":I
670 NEXT J
675 NEXT I
680 GOSUB 775
685 PRINT : INPUT "HIT 'RETURN' TO CONTINUE ";Y$: HOME
690 RETURN
695 HOME
700 PRINT "NEXT STEP TO BE PERFORMED":
705 PRINT " 1. EXIT PROGRAM"
710 PRINT " 2. MODIFY EXISTING DATA"
715 PRINT " 3. INPUT NEW DATA"
720 PRINT : PRINT : PRINT
725 PRINT " ENTER CHOICE # ": GET X
730 ON X GOTO 640, 875, 15
735 PRINT "CHOICE MUST BE 1, 2, 3, OR 4": GOTO 595
740 HOME: END
745 HOME: FOR X = 1 TO 5: PRINT : NEXT X
750 PRINT " PLANT DESIGN": PRINT : PRINT " PRODUCTION DEPARTMENT": PRINT " RELATIONSHIP CHART"
655 PRINT
660 PRINT "ITEMS DEPARTMENT"
665 PRINT "UNIVERSITY"
670 PRINT "OF"
675 PRINT "CENTRAL FLORIDA"
680 FOR X = 1 TO 3000: NEXT X: HOME
685 RETURN
690 STOP
695 FOR J = J TO NP:N(J) = 0
700 PRINT : PRINT : PRINT : PRINT
705 A$(1) = "QUANTITY ":A$(2) = "BATCH SIZE ":A$(3) = "DEPT.
# OF OPERATIONS 
710 FOR X = 1 TO 3: PRINT X;" ":A$(X): NEXT X
715 PRINT : PRINT : PRINT "ENTER 
# OF ITEM TO BE CHANGED ": PRINT 
: INPUT X
720 IF X = 1 THEN 735
725 IF X = 2 THEN 740
730 IF X = 3 THEN 745
735 PRINT : INPUT "ENTER NEW VALUE FOR QUANTITY ";Q(J); RETURN
740 PRINT : INPUT "ENTER NEW VALUE FOR BATCH SIZE ";B(J); RETURN
745 FOR I = 1 TO 25
750 PRINT "ENTER NEW DEPT. # OF 
OPERATION ";I;
755 INPUT ":P(J,I)
760 IF P(J,I) = 0 THEN RETURN
765 N(J) = N(J) + 1: NEXT I
770 RETURN
775 PRINT
780 PRINT "-----------------------------
-----------------------------
785 PRINT "REL. VAL. KEY"
790 PRINT "-----------------------------
-----------------------------
795 PRINT "6: ABSOLUTELY CLOSE 3 
: ORDINARY CLOSE"
PRINT "5: ESPECIALLY CLOSE 2 UNIMPORTANT"
PRINT "4: IMPORTANT 1 UNDESIRABLE"
RETURN
HOME
PRINT "RELATIONSH IF VALUES ARE": GOSUB 775
PRINT : PRINT : PRINT
INPUT "ENTER DEPT. # (FROM) " ; I
INPUT "ENTER DEPT. # (TO) " ; J
IF I > ND OR I < = 0 GOTO 8 55
IF J > ND OR J < = 0 GOTO 8 55
GOTO 860
PRINT : PRINT : PRINT "DEPT . # OUT OF RANGE": GOTO 830
INPUT "ENTER RELVAL " ; I, J
DD(I, J) = DD(I, J)
PRINT : PRINT "OVERWRITE HAS BEEN IMPLEMENTED": PRINT : GOTO 485
GOSUB 880 : GOTO 250
HOME : PRINT : PRINT : PRINT
PRINT "ENTER PART# FOR WHICH DATA": INPUT "TO BE MODIFIED (O TO CONTINUE) " ; J
IF J = 0 THEN 905
IF J > NP OR J < 0 THEN GOTO 910
GOSUB 135 : GOTO 880
RETURN
PRINT : PRINT "** PART# OUT OF RANGE **": PRINT : PRINT
GOTO 885
END
Case Example

A company consists of seven production departments (#1 through #7). Products are classified into five groups which have to pass through a specific operation sequence. The volume of production of each group, the bulk factor (Batch size or pieces/load), and the operation sequence are given in Table 2.1, Chapter II.

The Industrial Engineer needs to establish a relationship between the 7 departments so that he can arrange them in a way that satisfies the production requirements for all five groups of products.

After loading the program and typing RUN, the screen displays the name of the program, and the program will start accepting data through the users response to the questions posed. The program begins as follows:

JRUN

PLANT DESIGN
PRODUCTION DEPARTMENT RELATIONSHIP CHART
IEMS DEPARTMENT UNIVERSITY OF CENTRAL FLORIDA

ENTER # OF PARTS TO BE PRODUCED 5
ENTER # OF PRODUCTION DEPARTMENTS 7
This is followed by a brief statement of program instructions.

FOR EACH PART ENTER:

QUANTITY, BATCH SIZE, SEQUENCE
OF OPERATIONS

NOTE:
1- USE DEPT. #’S FOR OPERATIONS
2- USED DEPT. #’S CAN NOT EXCEED 7
3- ENTER ’0’ TO END OPERATIONAL SEQ.
4- # OF OPERATIONS CAN NOT EXCEED 50 PER PART

HIT RETURN WHEN READY

The required data is now entered.

PART # 1
ENTER QUANTITY 100
ENTER BATCH SIZE 7
ENTER DEPT. # FOR OPERATION 1 : 1
ENTER DEPT. # FOR OPERATION 2 : 5
ENTER DEPT. # FOR OPERATION 3 : 4
ENTER DEPT. # FOR OPERATION 4 : 3
ENTER DEPT. # FOR OPERATION 5 : 7
ENTER DEPT. # FOR OPERATION 6 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? Y
The opportunity to edit the data is given after entering data for each product group. The correction options are displayed:

1. QUANTITY
2. BATCH SIZE
3. DEPT. # OF OPERATIONS

Corrections are made by entering the number of the item to be changed and the corresponding value.

ENTER # OF ITEM TO BE CHANGED
?

ENTER NEW VALUE FOR QUANTITY 700
the remaining data is now entered

PART # 2
ENTER QUANTITY 12
ENTER BATCH SIZE 1
ENTER DEPT.# FOR OPERATION 1 : 1
ENTER DEPT.# FOR OPERATION 2 : 3
ENTER DEPT.# FOR OPERATION 3 : 2
ENTER DEPT.# FOR OPERATION 4 : 6
ENTER DEPT.# FOR OPERATION 5 : 7
ENTER DEPT.# FOR OPERATION 6 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? N

PART # 3
ENTER QUANTITY 1000
ENTER BATCH SIZE 20
ENTER DEPT.# FOR OPERATION 1 : 1
ENTER DEPT.# FOR OPERATION 2 : 5
ENTER DEPT.# FOR OPERATION 3 : 7
ENTER DEPT.# FOR OPERATION 4 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? N

PART # 4
ENTER QUANTITY 390
ENTER BATCH SIZE 30
ENTER DEPT.# FOR OPERATION 1 : 1
ENTER DEPT.# FOR OPERATION 2 : 2
ENTER DEPT.# FOR OPERATION 3 : 3
ENTER DEPT.# FOR OPERATION 4 : 4
ENTER DEPT.# FOR OPERATION 5 : 5
ENTER DEPT.# FOR OPERATION 6 : 6
ENTER DEPT.# FOR OPERATION 7 : 7
ENTER DEPT.# FOR OPERATION 8 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? N

PART # 5
ENTER QUANTITY 77
ENTER BATCH SIZE 7
ENTER DEPT.# FOR OPERATION 1 : 1
ENTER DEPT.# FOR OPERATION 2 : 5
ENTER DEPT.# FOR OPERATION 3 : 2
ENTER DEPT.# FOR OPERATION 4 : 3
ENTER DEPT.# FOR OPERATION 5 : 7
ENTER DEPT.# FOR OPERATION 6 : 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? N
While the computer is calculating the product REL values, the program will display:

YOUR COMPUTER IS CALCULATING PLEASE STAND BY

After the period in which computation takes place, a note is displayed to the user. The layout designer must supply the extreme values for the "Absolutely Necessary" relationship and the "Undesireable" relationship if pertinent. This is done by responding YES to the following questions:

NOTE:

- THE USER MAY SUPPLY CERTAIN RELVALS WHICH WILL AUTOMATICALLY OVERRIDE THE GENERATED VALUES

- RELVAL NO.1 AND NO.6 MUST BE SUPPLIED BY THE USER IF PERTINENT

DO YOU WANT TO SUPPLY ANY RELVALS?

(Y OR N) Y
A RELVAL KEY is now displayed and the data is input with the following format:

RELATIONSHIP VALUES ARE

<table>
<thead>
<tr>
<th>RELVAL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>ABSOLUTELY CLOSE</td>
</tr>
<tr>
<td>5</td>
<td>ESPECIALLY CLOSE</td>
</tr>
<tr>
<td>4</td>
<td>IMPORTANT</td>
</tr>
<tr>
<td>3</td>
<td>ORDINARY CLOSE</td>
</tr>
<tr>
<td>2</td>
<td>UNIMPORTANT</td>
</tr>
<tr>
<td>1</td>
<td>UNDESIRABLE</td>
</tr>
</tbody>
</table>

ENTER DEPT. # (FROM) 1
ENTER DEPT. # (TO) 7
ENTER RELVAL 6

A message is displayed to verify the modifications and the user is given the option to supply additional RELVALs.

OVERRIDE HAS BEEN IMPLEMENTED

DO YOU WANT TO SUPPLY ANY RELVALS?

(Y OR N) N

The Departmental Relationship Chart is now displayed

DEPARTMENTAL RELATIONSHIP CHART

BETWEEN DEPT. 1
AND DEPT. 1 RELVAL= 0
AND DEPT. 2 RELVAL= 2
AND DEPT. 3 RELVAL= 2
AND DEPT. 4 RELVAL= 2
AND DEPT. 5 RELVAL= 5
AND DEPT. 6 RELVAL= 2
AND DEPT. 7 RELVAL= 6
BETWEEN DEPT. 2
  AND DEPT. 1 RELVAL = 2
  AND DEPT. 2 RELVAL = 0
  AND DEPT. 3 RELVAL = 2
  AND DEPT. 4 RELVAL = 2
  AND DEPT. 5 RELVAL = 2
  AND DEPT. 6 RELVAL = 2
  AND DEPT. 7 RELVAL = 2

BETWEEN DEPT. 3
  AND DEPT. 1 RELVAL = 2
  AND DEPT. 2 RELVAL = 2
  AND DEPT. 3 RELVAL = 0
  AND DEPT. 4 RELVAL = 2
  AND DEPT. 5 RELVAL = 2
  AND DEPT. 6 RELVAL = 2
  AND DEPT. 7 RELVAL = 2

BETWEEN DEPT. 4
  AND DEPT. 1 RELVAL = 2
  AND DEPT. 2 RELVAL = 2
  AND DEPT. 3 RELVAL = 2
  AND DEPT. 4 RELVAL = 0
  AND DEPT. 5 RELVAL = 2
  AND DEPT. 6 RELVAL = 2
  AND DEPT. 7 RELVAL = 2

BETWEEN DEPT. 5
  AND DEPT. 1 RELVAL = 5
  AND DEPT. 2 RELVAL = 2
  AND DEPT. 3 RELVAL = 2
  AND DEPT. 4 RELVAL = 2
  AND DEPT. 5 RELVAL = 0
  AND DEPT. 6 RELVAL = 2
  AND DEPT. 7 RELVAL = 3

BETWEEN DEPT. 6
  AND DEPT. 1 RELVAL = 2
  AND DEPT. 2 RELVAL = 2
  AND DEPT. 3 RELVAL = 2
  AND DEPT. 4 RELVAL = 2
  AND DEPT. 5 RELVAL = 2
  AND DEPT. 6 RELVAL = 0
  AND DEPT. 7 RELVAL = 2
BETWEEN DEPT. 7
   AND DEPT. 1 RELVAL= 6
   AND DEPT. 2 RELVAL= 2
   AND DEPT. 3 RELVAL= 2
   AND DEPT. 4 RELVAL= 2
   AND DEPT. 5 RELVAL= 3
   AND DEPT. 6 RELVAL= 2
   AND DEPT. 7 RELVAL= 0
---------------------------------------------
   RELVAL KEY
---------------------------------------------
   6: ABSOLUTELY CLOSE 3: ORDINARY CLOSE
   5: ESPECIALLY CLOSE 2: UNIMPORTANT
   4: IMPORTANT 1: UNDESIRABLE

HIT 'RETURN' TO CONTINUE

finally, program options are now displayed

NEXT STEP TO BE PERFORMED :
    1. EXIT PROGRAM
    2. MODIFY EXISTING DATA
    3. INPUT NEW DATA

ENTER CHOICE #
Figure 4.1. Relationship chart summarizing the closeness relationship between the production departments using case example.
CHAPTER V

CORELAP

Introduction

CORELAP (Computerized Relationship Layout Planning) was originally presented by Lee and Moore [9]. This program is considered a construction type and it is not limited to production plants. CORELAP can be used to aid in the design of layouts where a relationship between plant departments was established. The minimum input requirements for CORELAP include:

1. Relationship Chart (REL Chart) between the departments, using standard closeness ratings (Table 5.1).
2. Departmental areas.
3. Weights for the REL Chart entries.

The REL Chart provides the basis for the order in which departments are arranged in the layout. The algorithm assigns numerical values to the closeness ratings and these values are added for each department to produce the total closeness rating for departments. This total represents the importance of each department in the layout with respect to other departments. The algorithm places the department having the largest total closeness rating is the middle of the layout.
A = Absolutely necessary.
E = Especially important.
I = Important.
O = Ordinary closeness desired.
U = Unimportant.
X = Undesireable.

The arrangement routine used in this program, originally presented by Hosni [4], is an adapted version of CORELAP. Specific assumptions are made which can be summarized as follows:

1. Departments are arranged in bays.
2. Departments are rectangular in shape, with its width equal to the bay width, and its length is computed based on department area.

This arrangement scheme places the department having the largest total closeness rating in the middle of the layout. The arrangement is expanded to the right and to the left of the "physical" middle, satisfying the "needs" of closeness as represented by the REL Chart. (see Figure 5.1).
Figure 5.1. Arrangement scheme which places most important department in the "physical" middle of the layout, and expands to the right and left, continuing in the next bays as shown.
This scheme is achieved by considering the plant bays as one continuous bay. The length is equal to the number of bays * area length. The algorithm determines the center and assigns the most important department. The algorithm keeps adding the different department lengths according to their closeness ratings, to the right and to the left. Each time there is an addition of a department either to the right or to the left, the bay is checked if it is full. If one side is filled, the arrangement continues on the other side. There is no way that the bays would not accommodate all the departments since the available space requirements are checked for sufficiency prior to the arrangement routine.

Advantages of CORELAP

1. This program takes into consideration the qualitative factors influencing the arrangement of departments as represented in the REL Chart.
2. CORELAP is a construction program, i.e., no initial layout is required.

Limitations of CORELAP

1. Original version of CORELAP is impractical with respect to the production plants because the final output design usually has "zig-zagging" boundaries in a way that it would be difficult to use material handling equipment *. However, the adapted version of CORELAP presented in this program eliminates this "zig-zagging" effect by constricting departments within boundaries of the bay.
2. CORELAP produces a single final layout, therefore no alternative designs are available for selection.

* The "zig-zagging" might be useful in satisfying the REL Chart requirements.
Program Documentation

Program Name: CORELAP (Computerized Relationship Layout Planning).

Program Function: CORELAP is a construction type program. It accepts a Relationship Chart between departments and produces a final layout.

Program Input: 1. Number of production departments.
2. REL values between departments expressed in numerical values (A=6, E=5, I=4, O=3, U=2, X=1).
3. Length of available area.
4. Width of available area.
5. Number of bays desired.
6. Departmental areas.

Program Logic: 1. Accept Input (REL Chart = N x N).
2. Display data and accept changes if any, if not, continue.
3. Department total closeness ratings are computed and displayed.

\[
\text{Total Closeness Rating} = \sum_{i=1}^{N} \sum_{j=1}^{N} \text{REL values} \quad \text{for } i = 1, 2, \ldots, N
\]

4. Departments are sorted according to their total closeness ratings.
5. Departments are arranged by placing the department having the largest total closeness rating in the middle of the layout. The algorithm keeps adding the different department lengths according to their closeness ratings, to the right and to the left. Each time there is an addition of a department either to the right or left, the bay is checked if it is full. If one side is filled, the arrangement continues on the other side.
6. Layout arrangement is displayed in terms of XY - coordinates and distance matrix. (Refer to Chapter III).

Program Output: Layout arrangement in terms of XY - coordinates for department areas and a rectilinear distance matrix.
Program Listing

10 REM **PLANT LAYOUT-DEPARTMENTAL RELATIONSHIP CHART
15 GOSUB 190
20 CLEAR
25 PRINT: PRINT: PRINT
30 INPUT "ENTER # OF PRODUCTION DEPARTMENTS "; ND
35 DIM DD(NO, ND)
40 PRINT
45 HOME
50 FOR RT = 1 TO 3: PRINT: NEXT RT
55 PRINT "INPUT THE RELVALS BETWEEN DEPARTMENTS"
60 PRINT: PRINT
65 PRINT "DEPARTMENTAL RELATIONSHIP CHART": PRINT "-----------------------------": PRINT
70 FOR I = 1 TO ND
75 PRINT "BETWEEN DEPT. ":I
80 FOR J = 1 TO ND
85 IF J = I GOTO 130
90 PRINT "AND DEPT. ":J
90 PRINT "RELVAL = ":DD(I, J)
95 IC = IC + 1
100 IF IC < 9 THEN 130
105 GOSUB 235
110 PRINT: PRINT "HIT 'RETURN' TO CONTINUE ": GET Y$
115 IC = 0
120 IF J = ND THEN HOME: GOTO 130
125 HOME: PRINT "BETWEEN DEPT. ":I
130 NEXT J
135 NEXT I
140 GOSUB 235
145 PRINT
150 INPUT "DO YOU WANT TO MAKE A NY CHANGES (Y OR N) "; Y$
155 IF Y$ = "N" GOTO 275
160 IF Y$ = "Y" GOTO 165
HOME : PRINT : PRINT
INPUT "CHANGE DATA FROM DEPT . " ; I2
INPUT " TO DEPT . " ; J2
INPUT "RELVAL = "; ID(I2, J2)
PRINT : PRINT " ** CHANGE IMPLEMENTED **" : GOTO 150
HOME : FOR X = 1 TO 5 : PRINT : NEXT X
PRINT "CORELA F"
PRINT "IEMS DEPARKMENT"
PRINT "UNIVERSITY"
PRINT "OF"
PRINT "CENTRAL FLORIDA"
FOR X = 1 TO 3000 : NEXT X : HOME
RETURN
PRINT
PRINT "-----------------------------"
PRINT "RELVAL KEY"
PRINT "-----------------------------"
PRINT "6: ABSOLUTELY CLOSE 3:
ORDINARY CLOSE"
PRINT "5: ESPECIALLY CLOSE 2:
UNIMPORTANT"
PRINT "4: IMPORTANT 1:
UNDESIRABLE"
RETURN
REM ** CRAFT **
FOR X = 1 TO 4 : PRINT : NEXT X
HOME : PRINT : PRINT
INPUT "ENTER TOTAL BAY LENGTH " ; BL : PRINT
INPUT "ENTER WIDTH OF AVAILABLE AREA " ; BW : PRINT
PRINT "**********************************************************************
**********************************************************************"
305 Z = BL * BW: PRINT "TOTAL SPACE AVAILABLE = "; Z; " SQ. UNIT S"
310 PRINT "***********************************************************************
315 INPUT "ENTER # OF BAYS DESIRED "; NB
320 PRINT
325 PRINT "***********************************************************************"
330 BS = BW / NB
335 PRINT "BAY SPAN = "; BS
340 PRINT "***********************************************************************
345 PRINT "HIT RETURN TO CONTINUE " "; GET Y$
350 DIM BAC(NB), BL(NB, 25), B(NB, 25)
355 DIM X1(NB, 25), Y1(NB, 25), X2(NB, 25), Y2(NB, 25)
360 DIM DA(50), IN(50), DL(50), DB(25), FG(10), FE(10)
365 DIM RG(50), SL(50), RF(50), LF(50), TL(50), TM(50), MT(50)
370 HOME; FOR X = 1 TO 5: PRINT 
375 BB = 0
380 PRINT "***********************************************************************
385 PRINT "FOR THE FOLLOWING SEQUENCE"
390 PRINT "ENTER : DEPT. #, AREA"
395 PRINT "***********************************************************************"
400 R1 = 0: R2 = 0: D1 = 0: F = 1: S = 0: BB = 0
405 FOR I = 1 TO NB: BAC(I) = 0: NEXT I
410 FOR I = 1 TO NB: DB(I) = 0: NEXT I
415 TT = 0
420 FOR I = 1 TO NB: FG(I) = 0: NEXT I
425 PRINT
430 PRINT "SEQUENCE, DEPT. #, AREA"
435 PRINT "-------------------------------"
440 FOR J = 1 TO ND
445 BB = BB + 1
450 PRINT "  ";'BB';: INPUT "
  ";IN(J),DA(J)
455 MT(J) = DA(J)
460 DL(J) = DA(J) / BS
465 TT = TT + DA(J)
470 IF TT > Z GOTO 1280
475 NEXT J
480 FOR I = 1 TO ND
485 FOR J = 1 TO ND
490 RG(I) = RG(I) + DI(I, J)
495 NEXT J
500 NEXT I
505 HOME : PRINT : PRINT
510 PRINT "***************
  "
515 PRINT " DEPARTMENT RATING"
520 PRINT "***************
  " : PRINT : PRINT
525 PRINT " DEPT.#  RATING"
530 PRINT "---------------------
  " : PRINT
535 FOR I = 1 TO ND
540 IN(I) = I
545 PRINT "  ";'IN(I);"
  ";RG(I)
550 NEXT I
555 PRINT : PRINT "HIT 'RETURN'
  TO CONTINUE" : GET Y$
560 HOME
565 N1 = ND - 1
570 FOR I = 1 TO N1
575 J = I + 1
580 IF RG(I) > = RG(J) GOTO 655
585 SG = RG(I)
590 RG(I) = RG(J)
595 RG(J) = SG
600 SQ = IN(I)
605 IN(I) = IN(J)
610 IN(J) = SQ
615 DA = DA(I)
620 DA(I) = DA(J);DA(J) = DQ
625 FOR JJ = 1 TO ND
630 SL(JJ) = DD(I, JJ)
635 DD(I, JJ) = DD(J, JJ)
640 DD(J, JJ) = SL(JJ)
645 NEXT JJ
650 GOTO 570
655 NEXT I
670 PRINT "SORTING BY RATING"
675 PRINT "******************************************************************************
680 PRINT ;PRINT ;PRINT
685 PRINT "DEPT.# RATING"
690 PRINT "---------------------------------------
-----" ;PRINT
695 FOR I = 1 TO ND
700 DL(I) = DA(I) / BS
705 PRINT "";IN(I);" ";RG(I)
710 NEXT I
715 PRINT ;PRINT "HIT 'RETURN'
TO CONTINUE " ;; GET Y$
720 HOME
725 LN = NB * BL
730 CN = LN / 2
735 I1 = 1
740 I2 = 1
745 LS = DL(1) / 2
750 RS = DL(1) / 2
755 LF(I1) = IN(1)
760 RF(I2) = IN(1)
765 FOR I = 2 TO ND STEP 2
770 J = I
775 IF LS >= CN GOTO 800
780 LS = LS + DL(J)
785 I1 = I1 + 1
790 LF(I1) = IN(J)
795 J = J + 1
800 IF RS >= CN GOTO 775
805 IF J >= ND GOTO 825
810 RS = RS + DL(J)
815 I2 = I2 + 1
820 RF(I2) = IN(J)
825 NEXT I
830 FOR I = 1 TO ND
835 IN(I) = LF(I1)
840 I1 = I1 - 1
845 IF I1 = 0 GOTO 855
850 NEXT I
855 J1 = 1
860 FOR J = I TO ND
865 IN(J) = RF(J1)
870 J1 = J1 + 1
875 IF J1 > I2 GOTO 885
880 NEXT J
885 FOR I = 1 TO ND
890 FK = DN(I)
895 TM(I) = MT(FK)
900 DA(I) = TM(I)
905 NEXT I
910 PRINT : PRINT : PRINT
915 PRINT "******************************************
920 PRINT "ARRANGEMENT-HIGHEST RATING IN CENTER"
925 PRINT "******************************************"
930 PRINT : PRINT : PRINT
935 PRINT " DEPT.# LENG TH"
940 PRINT "---------------------
945 FOR I = 1 TO ND
950 DL(I) = DA(I) / BS
955 PRINT " ";DN(I);"
960 NEXT I
965 PRINT : PRINT "HIT 'RETURN' TO CONTINUE "; GET Y$
970 FOR I = 1 TO NB
975 R1 = 0: D1 = 0
980 FOR J = F TO ND
985 S = S + 1
990 DL(J) = DA(J) / BS
995 BAC(I) = BAC(I) + DL(J)
1000 IF BAC(I) < BL GOTO 1015
1005 IF BAC(I) = BL GOTO 1055
1010 IF BAC(I) > BL GOTO 1095
1015 R1 = BL - BAC(I)
1020 IF R1 < 10 GOTO 1040
1025 BL(I,S) = DL(J)
1030 B(I,S) = DN(J)
1035 GOTO 1050
1040 DL(J) = DL(J) + R1
1045 GOTO 1055
1050 NEXT J
1055 BL(I,S) = DL(J)
1060 B(I,S) = DN(J)
1065 DB(I) = J - F + 1
1070 IF FG(I) = 1 THEN DB(I) = D(B(I)) + 1
1075 F = J + 1
1080 FB = 0
1085  S = 0
1090  GOTO 1190
1095  R1 = BAC(I) - BL
1100  IF R1 < 10 GOTO 1180
1105  R2 = DL(J) - R1
1110  BL(I,S) = R2
1115  B(I,S) = DN(J)
1120  FB(I) = 1
1125  DB(I) = J - F + 1
1130  IF FG(I) = 1 THEN DB(I) = B
    R(I) + 1
1135  M = I + 1
1140  FG(M) = 1
1145  D1 = DN(J)
1150  BAC(M) = BAC(M) + R1
1155  BL(M,1) = R1
1160  B(M,1) = D1
1165  S = 1
1170  F = J + 1
1175  GOTO 1190
1180  DL(J) = DL(J) - R1
1185  GOTO 1055
1190  NEXT I
1195  PRINT ; PRINT
1200  HOME
1205  FOR I = 1 TO NB
1210  IF BL(I,1) = 0 GOTO 1220
1215  GOTO 1270
1220  J1 = DB(I)
1225  J1 = J1 - 1
1230  FOR J = 1 TO J1
1235  J2 = J + 1
1240  BL(I,J) = BL(I,J2)
1245  B(I,J) = B(I,J2)
1250  NEXT J
1255  DB(I) = DB(I) - 1
1260  M = I - 1
1265  DB(M) = DB(M) - 1
1270  NEXT I
1275  GOTO 1295
1280  PRINT ; PRINT ; PRINT "****
***********
****" 1285  PRINT "THE TOTAL AVAILABLE
SPACE REQUIREMENTS" ; PRINT "HAVE BEEN EXCEEDED BY ";TT -
Z" SQ.UNITS
1290  PRINT "***********
***********" ; PRINT
1295 HOME : PRINT : PRINT : PRINT : PRINT
: PRINT
1300 PRINT " SUBROUTINE CAL
CULATING"
1305 PRINT
1310 PRINT " XY-COORDINA
TES"
1315 FOR X = 1 TO 3000: NEXT X
1320 HOME
1325 FOR I = 1 TO NB
1330 IF I = 1 GOTO 1355
1335 IF I = 3 GOTO 1355
1340 F = I / 2
1345 F = F * 2
1350 IF F = 1 GOTO 1405
1355 X1(I,1) = 0
1360 N1 = DB(I)
1365 X(1(I,1) = BL(I,1) / 2
1370 FOR J = 1 TO N1
1375 X2(I,J) = X1(I,J) + BL(I,J)
1380 J1 = J + 1
1385 IF J = N1 GOTO 1395
1390 X1(I,J1) = X2(I,J)
1395 NEXT J
1400 GOTO 1450
1405 N1 = DB(I)
1410 X2(I,1) = BL
1415 FOR J3 = 1 TO N1
1420 X1(I,J3) = X2(I,J3) - BL(I,J 
1425 J2 = J - 1
1430 J4 = J3 + 1
1435 IF J3 = N1 GOTO 1445
1440 X2(I,J4) = X1(I,J3)
1445 NEXT J3
1450 NEXT I
1455 Y1(1,1) = BW
1460 FOR I = 1 TO NB
1465 N1 = DB(I)
1470 FOR J = 1 TO N1
1475 Y1(I,J) = Y1(I,1)
1480 Y2(I,J) = Y1(I,J) - BS
1485 NEXT J
1490 IF I = NB GOTO 1505
1495 J2 = I + 1
1500 Y1(I2,1) = Y2(I,1)
1505 NEXT I
1510 FOR I = 1 TO NB
1515 J2 = DB(I)
1520 FOR J = 1 TO J2
1525 \( XC(I, J) = (X1(I, J) + ((BL(I, J)) / 2.0)) \)
1530 NEXT J
1535 NEXT I
1540 HALF = 0.5 * BS
1545 YC(1, 1) = Y1(1, 1) - HALF
1550 J6 = DB(I)
1555 FOR J = 1 TO J6
1560 YC(1, J) = YC(1, 1)
1565 NEXT J
1570 FOR I = 2 TO NB
1575 J6 = DB(I)
1580 FOR J = 1 TO J6
1585 I2 = I - 1
1590 YC(I, J) = YC(I2, 1) - BS
1595 NEXT J
1600 NEXT I
1605 HOME : PRINT : PRINT
1610 PRINT """" XY-COORDINATES CHART"
1615 PRINT """" -----------------------------
1620 PRINT "SEG \""DEP \""X1 \""Y1 \""X2 \""Y2 \""XC \""YC"
1625 PRINT """" -----------------------------
1630 BB = 0
1635 FOR I = 1 TO NB
1640 J1 = DB(I)
1645 FOR J = 1 TO J1
1650 BB = BB + 1
1655 Y1(I, J) = ABS (Y1(I, J) - BW)
1660 Y2(I, J) = ABS (Y2(I, J) - BW)
1665 YC(I, J) = ABS (YC(I, J) - BW)
1670 YC(1, J) = YC(1, 1)
1690 NEXT J
1695 NEXT I
1700 M = I + 1
1705 IF M > NB GOTO 1805
1715 NEXT I
1720 GOTO 1805
1725 M = 0
1730 FOR I = 1 TO NB
1735 J1 = DB(I)
1740 M = I + 1
1745 IF M > NB GOTO 1805
1750 IF BL(I, J1) > BL(M, 1) GOTO 1765
1755 XC(I, J1) = XC(M, 1); YC(I, J1) = YC(M, 1)
1760 GOTO 1770
1765 XC(M, 1) = XC(I, J1); YC(M, 1) = YC(I, J1)
1770 NEXT I
1775 GOTO 1715
1780 IF M > NB GOTO 1805
1785 IF BL(I, J1) > BL(M, 1) GOTO 1795
1790 GOTO 1800
1795 XC(M, 1) = XC(I, J1); YC(M, 1) = YC(I, J1)
1800 NEXT I
1805 PRINT; PRINT "HIT 'RETURN' TO CONTINUE"; : GET Y$:
1810 HOME
1815 FOR I = 1 TO ND
1820 FOR J = 1 TO ND
1825 DS(I, J) = 99999
1830 NEXT J
1835 NEXT I
1840 FOR I = 1 TO NB
1845 J2 = DB(I)
1850 FOR J = 1 TO J2
1855 FOR I1 = 1 TO NB
1860 J3 = DB(I)
1865 FOR J1 = 1 TO J3
1870 J5 = B(I, J)
1875 J4 = B(I1, J1)
1880 D1 = ABS(XC(I, J) - XC(I1, J1))
1885 D1 = D1 + ABS(YC(I, J) - YC(I1, J1))
1890 IF D1 < = DS(J5, J4) THEN D$ = DS(J5, J4) = D1
1895 NEXT J1
1900 NEXT I1
1905 NEXT J
1910 NEXT I
1915 PRINT: PRINT: PRINT
1920 PRINT : PRINT "DISTANCE-MATRIX"
1925 PRINT "---------------------"

1930 J = 0
1935 FOR F = 1 TO ND
1940 FOR F1 = 1 TO ND
1945 IF DS(F,F1) < DS(F1,F) THEN
       DS(F1,F) = DS(F,F1)
1950 IF DS(F1,F) < DS(F,F1) THEN
       DS(F,F1) = DS(F1,F)
1955 J = J + 1
1960 A = A + 5
1965 IF JP = 1 GOTO 1975
1970 PRINT DS(F,F1); TAB(A); 
1975 IF J = ND THEN J = 0: PRINT

1980 NEXT F1
1985 A = 0
1990 NEXT F
1995 PRINT : PRINT : PRINT "HIT "RETURN" TO CONTINUE ": GET Y$
2000 HOME : PRINT : PRINT : PRINT "$ DO YOU WANT TO EXIT PROGRAM (Y OR N)? "; B$: IF
2005 GOTO 10 THEN 2010
2010 END
Case Example

A company consists of five production departments (#1 through #5). The Industrial Engineer established a relationship between the five departments as follows:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>From</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>5</td>
<td>0</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>6</td>
<td>6</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>6</td>
<td>4</td>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>
The Industrial Engineers desire is to construct a layout arrangement to satisfy this Relationship Chart.

After loading the program and typing RUN, the screen displays the following:

```
RUN
```

The required data is now entered by answering the pertinent questions:

```
ENTER # OF PRODUCTION DEPARTMENTS 5

INPUT THE RELVALS BETWEEN DEPARTMENTS

DEPARTMENTAL RELATIONSHIP CHART

BETWEEN DEPT. 1
AND DEPT. 2 RELVAL = 5
AND DEPT. 3 RELVAL = 3
AND DEPT. 4 RELVAL = 4
AND DEPT. 5 RELVAL = 6
```
BETWEEN DEPT. 2
   AND DEPT. 1 RELVAL = 5
   AND DEPT. 3 RELVAL = 5
   AND DEPT. 4 RELVAL = 6
   AND DEPT. 5 RELVAL = 6
BETWEEN DEPT. 3
   AND DEPT. 1 RELVAL = 3
   AND DEPT. 2 RELVAL = 5
   AND DEPT. 4 RELVAL = 6
   AND DEPT. 5 RELVAL = 4
BETWEEN DEPT. 4
   AND DEPT. 1 RELVAL = 4
   AND DEPT. 2 RELVAL = 6
   AND DEPT. 3 RELVAL = 6
   AND DEPT. 5 RELVAL = 5
BETWEEN DEPT. 5
   AND DEPT. 1 RELVAL = 6
   AND DEPT. 2 RELVAL = 6
   AND DEPT. 3 RELVAL = 4
   AND DEPT. 4 RELVAL = 5

-------------------------------------
   RELVAL KEY
   -------------------------------------
   6: ABSOLUTELY CLOSE  3: ORDINARY CLOSE
   5: ESPECIALLY CLOSE  2: UNIMPORTANT
   4: IMPORTANT        1: UNDESIRABLE

DO YOU WANT TO MAKE ANY CHANGES? (Y OR N)? N
The dimensions of the space available are now entered

ENTER TOTAL BAY LENGTH 100
ENTER WIDTH OF AVAILABLE AREA 50

and the total space available is computed.

TOTAL SPACE AVAILABLE = 5000 SQ. UNITS

The number of bays desired are now entered

ENTER # OF BAYS DESIRED 2

and the bay span to calculated.

BAY SPAN = 25
The remaining data is now entered.

```
<table>
<thead>
<tr>
<th>SEQUENCE, DEPT, AREA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
</tbody>
</table>
```

The total closeness ratings are now computed and displayed.

```
<table>
<thead>
<tr>
<th>DEPT,</th>
<th>RATING</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>18</td>
</tr>
<tr>
<td>2</td>
<td>22</td>
</tr>
<tr>
<td>3</td>
<td>18</td>
</tr>
<tr>
<td>4</td>
<td>21</td>
</tr>
<tr>
<td>5</td>
<td>21</td>
</tr>
</tbody>
</table>
```
The departments are sorted according to their total closeness ratings and then displayed to the user.

**************************** SORTING BY RATING ****************************

<table>
<thead>
<tr>
<th>DEPT. #</th>
<th>RATING</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>22</td>
</tr>
<tr>
<td>4</td>
<td>21</td>
</tr>
<tr>
<td>5</td>
<td>21</td>
</tr>
<tr>
<td>1</td>
<td>18</td>
</tr>
<tr>
<td>3</td>
<td>18</td>
</tr>
</tbody>
</table>

The program now arranges the departments on the basis of total closeness ratings.

**************************** ARRANGEMENT-HIGHEST RATING IN CENTER ****************************

<table>
<thead>
<tr>
<th>DEPT. #</th>
<th>LENGTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>28</td>
</tr>
<tr>
<td>4</td>
<td>72</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
</tr>
<tr>
<td>5</td>
<td>28</td>
</tr>
<tr>
<td>3</td>
<td>32</td>
</tr>
</tbody>
</table>
While computing the XY-Coordinates, the program will display the following:

**SUBROUTINE CALCULATING**

**XY-COORDINATES**

After the period in which computation takes place, the results are displayed.

**XY-COORDINATES CHART**

<table>
<thead>
<tr>
<th>SEQ</th>
<th>DEP</th>
<th>X1</th>
<th>Y1</th>
<th>X2</th>
<th>Y2</th>
<th>XC</th>
<th>YC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>28</td>
<td>25</td>
<td>14</td>
<td>12.5</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>28</td>
<td>28</td>
<td>100</td>
<td>25</td>
<td>64</td>
<td>12.5</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>60</td>
<td>25</td>
<td>100</td>
<td>50</td>
<td>80</td>
<td>37.5</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>32</td>
<td>25</td>
<td>60</td>
<td>50</td>
<td>46</td>
<td>37.5</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>0</td>
<td>25</td>
<td>32</td>
<td>50</td>
<td>16</td>
<td>37.5</td>
</tr>
</tbody>
</table>

**DISTANCE-MATRIX**

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>91</th>
<th>27</th>
<th>50</th>
<th>57</th>
</tr>
</thead>
<tbody>
<tr>
<td>91</td>
<td>0</td>
<td>64</td>
<td>41</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>64</td>
<td>0</td>
<td>73</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>41</td>
<td>73</td>
<td>0</td>
<td>43</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>34</td>
<td>30</td>
<td>43</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Finally program options are now displayed.

**DO YOU WANT TO EXIT PROGRAM, (Y OR N)? Y**
CHAPTER VI
LAYOUT EVALUATION

Introduction

It is the purpose of this chapter to present a computer program written in Interactive Basic that evaluates a layout arrangement and provides guidelines for rearrangement with respect to satisfying a relationship chart.

The layout evaluation technique was developed by Hosni [4] to guide the layout designer in generating more efficient layout designs.

A product equivalent relationship chart between the production departments is generated by accepting product data. The criteria considered in creating these relationships are product volume, material flow, product design, and sequence of processing. The product REL Chart represents a measure of activity relationships by performing pairwise evaluations of the importance for two activities to be located together in the layout.

The closeness relationship between the product departments are assigned the following ratings:
<table>
<thead>
<tr>
<th>Relationship</th>
<th>Closeness Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolute necessary</td>
<td>6</td>
</tr>
<tr>
<td>Especially important</td>
<td>5</td>
</tr>
<tr>
<td>Important</td>
<td>4</td>
</tr>
<tr>
<td>Ordinary closeness</td>
<td>3</td>
</tr>
<tr>
<td>Unimportant</td>
<td>2</td>
</tr>
<tr>
<td>Undesireable</td>
<td>1</td>
</tr>
</tbody>
</table>

The layout designer must supply the extreme values for the "Absolutely Necessary" relationship and the "Undesireable relationship.

A layout REL Chart is formed using equivalent measures to those used in generating the Product REL Chart. The algorithm considers the adjacent departments as having a closeness rating of "6"; the departments separated by one department in between, a rating of "5"; those departments separated by two departments in between, a rating of "4"; etc. A demonstrating example is shown in Figure 6.1.
1. Layout Arrangement.

2. Translation as Input Data.

<table>
<thead>
<tr>
<th>Dept.</th>
<th>Adjacents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>5,4</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>2,3</td>
</tr>
<tr>
<td>5</td>
<td>1,2</td>
</tr>
</tbody>
</table>

3. Equivalent Layout REL Chart.

```
 1 2 3 4 5
1 0 5 3 4 6
2 5 0 5 6 6
3 3 5 0 6 4
4 4 6 6 0 5
5 6 6 4 5 0
```

Figure 6.1. Translation of layout arrangement and generation of an equivalent layout REL chart.
This algorithm also scores the arrangement, analyzing it and producing an efficiency value. This is achieved by comparing both the layout relationship chart and the product relationship chart. Grading is performed by subtracting the two matrices of the relationship and scoring the difference as follows:

<table>
<thead>
<tr>
<th>Difference (or negative)</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>Error</td>
</tr>
</tbody>
</table>

The grades of all the relationships are added together to produce a total score. Once the evaluation scoring has been determined, suggestions to generate alternative layout designs and the corresponding maximum increase in efficiency can be deduced.
Program Documentation

Program Name: Layout Evaluation.

Program Function: Evaluates present layout design and identifies those departments which if become more adjacent, will result in a more efficient layout.

Program Input: 1. Number of production departments.
2. REL values with respect to the product.
3. The corresponding adjacent departments for each department. (See case example).

Program Logic: 1. Accept input.
2. Display data, make corrections if any, if not, continue.
3. Relationship chart with respect to the layout is formed.
   - adjacent departments have a closeness rating of "6".
   - departments separated by one department in between a rating of "5".
   - departments separated by two departments in between, a rating of "4"; etc. (See Figure 6.1.)
4. Subtract product equivalent relationship chart from layout equivalent relationship chart to produce an evaluation scoring between departments.
   - difference of 0 (or negative), a grade of "5".
   - difference of 1, a grade of "4".
   - difference of 2, a grade of "3"; etc.
   - difference of 5 or greater produces an error.
5. The grades of all the relationships are added together to produce a total score. Once the evaluation scoring has been determined, suggestions of which departments should become more adjacent to provide a more efficient layout design can be deduced.

Program Output: 1. Relationship Chart with respect to the layout.
2. Evaluation scoring for each department.
3. Recommended improvements.
102

Program Listing

10 REM **PLANT LAYOUT EVALUATION**
   **N**
15 GOSUB 1030
20 HOME
25 PRINT : PRINT : INPUT "DO YOU
   NEED INSTRUCTIONS (Y OR N)
   ": Y$
30 IF Y$ = "Y" THEN GOSUB 45: GOTO 120
35 IF Y$ = "N" THEN 120
40 GOTO 25
45 HOME : PRINT : PRINT
50 PRINT "THIS PROGRAM IS DESIGNED TO EVALUATE"
55 PRINT "A DEPARTMENTAL ARRANGEMENT (LAYOUT)"
60 PRINT "WITH RESPECT TO SATISFYING A"
65 PRINT "RELATIONSHIP CHART TO BE SUPPLIED": PRINT "BY THE
   USER. EVALUATION CRITERIA": PRINT "IS THE PROXIMITY BETWEEN DEPARTMENTS."": PRINT
70 PRINT "PROGRAM ACCEPTS A RELATIONSHIP CHART": PRINT "USING THE FOLLOWING 'RELVAL'S:"
75 PRINT "RELVAL KEY"
80 PRINT "6: ABSOLUTELY CLOSE 3: ORDINARY CLOSE"
85 PRINT "5: ESPECIALLY CLOSE 2: UNIMPORTANT"
90 PRINT "4: IMPORTANT 1: UNDESIRABLE"
95 PRINT
100 PRINT "IT ACCEPTS DATA FOR THE LAYOUT IN": PRINT "THE FORM OF DEPT.# AND THE #S OF"
   : PRINT "DEPTS. ADJACENT TO IT."": PRINT
105 PRINT "PROGRAM SCORES THE LAYOUT AND": PRINT "RECOMMENDS IMPROVEMENTS."
110 PRINT : PRINT "HIT 'RETURN'
   WHEN READY": GET A$
115  RETURN
120  HOME
125  FOR I = 1 TO 8: PRINT : NEXT I
130  INPUT "ENTER THE # OF DEPARTMENTS":ND
135  DIM D(ND,ND),DD(ND,ND),E(ND,21)
140  HOME
145  FOR I = 1 TO ND
150  J1 = I + 1
155  IF J1 > ND AND I = ND THEN 25
160  PRINT "ENTER RELVAL BETWEEN DEPT. ":I
165  FOR J = J1 TO ND
170  PRINT "AND DEPT. ":J;" = "; ; INPUT DD(I,J)
175  IF DD(I,J) > 6 OR DD(I,J) < 1 THEN PRINT "RELVAL SHOULD BE BETWEEN 1 & 6"; PRINT "REENTER": GOTO 170
180  IC = IC + 1
185  IF IC < 16 THEN 215
190  GOSUB 1115
195  PRINT : PRINT "HIT 'RETURN' TO CONTINUE ": GET Y$
200  IC = 0
205  IF J = ND THEN HOME : GOTO 215
210  HOME : PRINT "ENTER RELVAL BETWEEN DEPT. ":I
215  NEXT J
220  NEXT I
225  GOSUB 1115
230  PRINT : PRINT "HIT 'RETURN' TO CONTINUE ": GET Y$
235  HOME : PRINT : PRINT
240  REM ** RELATIONSHIP CHART W.R.T. ADJACENT DEPTS.
245  PRINT "INPUT DAT A FOR": PRINT " LAYOUT TO BE EVALUATED"
250  PRINT : PRINT
255  PRINT " FOR EACH DEPT., ENTER THE # OF": PRINT " CORRESPONDING ADJACENT DEPTS."
260 PRINT : PRINT
265 PRINT "NOTE:"; PRINT "-----"
    PRINT : PRINT "   NO. 0
F ADJACENT DEPTS. NOT TO"; PRINT
    PRINT "   EXCEED 20."; PRINT : PRINT
    "   ENTER '0' TO END SEQUENCE OF"; PRINT " ADJAC
    ENT DEPTS."
270 PRINT : PRINT "HIT 'RETURN'
TO CONTINUE"; GET Y$; HOME
275 FOR I = 1 TO ND
280 HOME
285 PRINT "FOR DEPT. # "; I
290 FOR J = 1 TO 20
295 PRINT "   ADJACENT DEPT. "; J:" IS ";" INPUT E(I,J)
300 IF E(I,J) > ND THEN PRINT "
   DEPT.# CAN NOT EXCEED ";ND
    PRINT "REENTER": GOTO 295
305 IF E(I,J) = I THEN PRINT "D
   EPT. CANNOT BE ADJACENT TO I
   TSELF"; PRINT "REENTER": GOTO 295
310 IF E(I,J) = 0 THEN 325
315 NEXT J
320 PRINT : PRINT
325 PRINT : INPUT "DO YOU WANT T
O MAKE CHANGES, (Y OR N)%"; B
    $; IF B$ = "Y" THEN GOSUB 1
329
330 HOME
335 NEXT I
340 PRINT : PRINT : PRINT : PRINT
   "   YOUR COMPUTER IS CALCULA
   TING"
345 FOR I = 1 TO ND
350 FOR J = 1 TO 20
355 D(I,J) = 0
360 NEXT J
365 NEXT I
370 FOR I = 1 TO ND
375 FOR J = 1 TO 21
380 J1 = E(I,J)
385 IF J1 = I GOTO 545
390 IF J1 = 0 GOTO 550
395 IF D(I,J1) > 6 GOTO 405
400 D(I,J1) = 6
405 FOR K = 1 TO 21
410 K1 = E(J1,K)
415 IF K1 = 0 GOTO 545
420 IF K1 = 1 GOTO 540
425 IF D(I,K1) > 5 GOTO 435
430 D(I,K1) = 5
435 FOR L = 1 TO 21
440 L1 = E(K1,L)
445 IF L1 = 0 GOTO 540
450 IF L1 = 1 GOTO 535
455 IF D(I,L1) > 4 GOTO 465
460 D(I,L1) = 4
465 FOR M = 1 TO 21
470 M1 = E(L1,M)
475 IF M1 = 0 GOTO 535
480 IF M1 = 1 GOTO 530
485 IF D(I,M1) > 3 GOTO 495
490 D(I,M1) = 3
495 FOR T = 1 TO 21
500 T1 = E(M1,T)
505 IF T1 = 0 GOTO 520
510 IF T1 = 1 GOTO 525
515 IF D(I,T1) > 2 GOTO 525
520 D(I,T1) = 2
525 NEXT T
530 NEXT M
535 NEXT L
540 NEXT K
545 NEXT J
550 NEXT I
555 FOR I = 1 TO ND
560 FOR J = 1 TO ND
565 IF I = J THEN 575
570 IF D(I,J) = 0 THEN D(I,J) = 2
575 NEXT J
580 NEXT I
585 REM ** SYMMETRY**
590 FOR I = 1 TO ND
595 FOR J = 1 TO ND
600 IF D(I,J) > D(J,I) THEN D(J,I) = D(I,J)
605 IF D(J,I) > D(I,J) THEN D(I,J) = D(J,I)
610 NEXT J
615 NEXT I
620 GOSUB 625: GOTO 645
625 HOME
630 PRINT: PRINT
635 PRINT "LAYOUT EQUIVALENT": PRINT "RELATIONSHIP CHART": PRINT "--------------------": PRINT
640 RETURN
645 IC = 0
650 FOR I = 1 TO ND
655 PRINT "BETWEEN DEPT.# ":I
660 FOR J = 1 TO ND
665 IF I = J THEN 680
670 PRINT "AND DEPT.# ":J:" LAYREL= ":D(I,J)
675 IC = IC + 1
680 IF IC = 12 THEN PRINT: PRINT 
"HIT 'RETURN' TO CONTINUE": GET Y$: IC = 0: GOSUB 625: PRINT 
"BETWEEN DEPT.# ":I
685 NEXT J
690 NEXT I
695 PRINT: PRINT "HIT 'RETURN' 
TO CONTINUE": GET Y$
700 HOME
705 REM ** RELATIONSHIP CHART EVALUATION PROCESS
710 REM ** SUBTRACT REL. CHART W. R.T. PRODUCT - REL. CHART W. R.T. ADJACENT DEPTS.
715 FOR I = 1 TO ND - 1
720 FOR J = I + 1 TO ND
725 IF D(D(I,J) = 6 AND D(I,J) <
730 IF D(D(I,J) = 1 AND D(I,J) =
735 NEXT J
740 NEXT I
745 FOR I = 1 TO ND - 1
750 FOR J = I + 1 TO ND
755 X = D(D(I,J) - D(I,J)
760 NEXT J
765 NEXT I
PRINT : PRINT : PRINT "EVALUATION SCORING": PRINT "--------------------------": PRINT

L = 0: LT = 0
IC = 0
N1 = ((ND * ND) - ND) / 2
DIM JX(5), ME(5, N1, 2)
FOR I1 = 1 TO 5; JX(I1) = 0: NEXT I1
FOR I = 1 TO ND - 1
FOR J = I + 1 TO ND
X = DD(I, J) - D(I, J)
L = L + 1
LS = 0
IF X < = 0 THEN LS = 5: GOTO 845
FOR I1 = 1 TO 4
IF X = I1 THEN GOSUB 880: GOTO 845
NEXT I1
PRINT "FROM DEPT ": I: " TO DEPT ": J: " EVAL SCORE= ": LS
IC = IC + 1
IF IC = 11 THEN PRINT : PRINT : INPUT "HIT RETURN TO CONTINUE": Y*: IC = 0
LT = LT + LS
NEXT J
NEXT I
GOTO 905
REM **SUBROUTINE EFFICIENCY** *
IF X = I1 THEN LS = (5 - I1)
JD = LS + 1: JX(JD) = JX(JD) + 1
J1 = JX(JD)
ME(JD, J1, 1) = I: ME(JD, J1, 2) = J
RETURN
L = 5 * L
PRINT : PRINT "A TOTAL EVALUATION SCORE OF ": LT: PRINT :
PRINT "OUT OF A POSSIBLE MAXIMUM OF ": L: PRINT : PRINT "WAS DETERMINED."
915 PRINT : PRINT : INPUT "HIT R
RETURN TO CONTINUE" : Y$ : HOME

920 REM ** DISPLAY OF DATA **
925 FOR I = 1 TO 5
930 Z = 0 : R = 0
935 IF JX(I) = 0 THEN 980
940 HOME : PRINT : PRINT "IF THE
FOLLOWING DEPTS.:" : PRINT : PRINT
"BECAME CLOSER ---": PRINT :
PRINT
945 FOR J = 1 TO JX(I)
950 PRINT : PRINT "
DEPT
" ; ME(I, J, 1) ; 
& DEPT. " ; ME
(I, J, 2)
955 NEXT J
960 Z = 6 - I : R = R + Z
965 PRINT : PRINT "THIS WOULD IN
CREASE THE TOTAL": PRINT : PRINT
"EVALUATION SCORE TO "; LT +
(R * JX(I))
970 PRINT : PRINT 
"OUT OF A POSS
IBLE MAXIMUM OF "; L : PRINT
975 PRINT : PRINT : INPUT "HIT R
RETURN TO CONTINUE" : Y$
980 NEXT I
985 HOME : PRINT : PRINT : PRINT
: PRINT " NEXT STEP TO BE PE
RFORMED"
990 PRINT " 1. EXIT PROGRAM"
995 PRINT " 2. INPUT NEW DAT
A"
1000 PRINT : PRINT : PRINT " E
NTER CHOICE # ": GET X
1005 ON X GOTO 1020, 10
1010 PRINT "CHOICE MUST BE 1 OR
2": GOTO 985
1015 END
1020 HOME : END
1025 HOME : CLEAR : GOTO 10
1030 HOME : FOR X = 1 TO 5 : PRINT :
NEXT X
1035 PRINT " PLANT DE
SIGN"
PRINT " LAYOUT EVALUATION"
PRINT " ITEMS DEPARTMENT"
PRINT " UNIVERSITY"
PRINT " DEPARTMENT..."
PRINT " 1040 "
PRINT " 1045 "
PRINT " 1050 "
PRINT " 1055 "
PRINT " 1060 "
FOR X = 1 TO 3000: NEXT X: HOME
RETURN
STOP
FOR I = 1 TO N: HOME
FOR J = 1 TO 12
INPUT " NEW ADJACENT DEPT. = " ; E(I, J)
IF E(I, J) = 0 THEN RETURN
N%(I) = N%(I) + 1
NEXT J
RETURN
PRINT : INPUT " DO YOU WANT TO MAKE ANY CHANGES (Y OR N) ": Y$
IF Y$ = "N" GOTO 1150
IF Y$ < > "Y" THEN 1115
INPUT " CHANGE RELVOR BET. DEPT. ": I
INPUT " AND DEPT. ": J
INPUT " TO ": DI(I, J)
PRINT " *** CHANGE IMPLEMENTED ***": GOTO 1115
RETURN
HOME
FOR T = 1 TO 5: PRINT : NEXT T
PRINT " DEPT. #: ": I:" AND DEPT. #: ": J: PRINT " SHOULD BE ADJACENT IN, ANY": PRINT " LAYOUT ARRANGEMENT.": PRINT " THE LAYOUT IN HAND DOES NOT": PRINT " FULFILL THIS REQUIREMENT."
1170 PRINT: PRINT: PRINT "HIT 'RETURN' TO CONTINUE": GET Y$
1175 RETURN
1180 HOME
1185 FOR T = 1 TO 5: PRINT: NEXT T
1190 PRINT " DEPT.# " ; I ; " AN I DEPT.# " ; J ; PRINT " CANNOT BE ADJACENT IN ANY" : PRINT " LAYOUT ARRANGEMENT." : PRINT " THE LAYOUT IN HAND IOE S NOT" : PRINT " FULFILL T HIS REQUIREMENT."
1195 PRINT: PRINT: PRINT "HIT 'RETURN' TO CONTINUE": GET Y$
1200 RETURN
Case Example

A company consists of five production departments (#1 through #5).

The Industrial Engineer needs to evaluate a departmental layout with respect to satisfying a given relationship chart so that he can rearrange them with respect to satisfying production requirements.

After loading the program and typing RUN, the screen displays the following:

RUN

PLANT DESIGN
LAYOUT EVALUATION

IEMS DEPARTMENT
UNIVERSITY
OF
CENTRAL FLORIDA

This is followed by a brief statement of program instruction.

DO YOU NEED INSTRUCTIONS (Y OR N) Y

THIS PROGRAM IS DESIGNED TO EVALUATE A DEPARTMENTAL ARRANGEMENT (LAYOUT) WITH RESPECT TO SATISFYING A RELATIONSHIP CHART TO BE SUPPLIED BY THE USER. EVALUATION CRITERIA IS THE PROXIMITY BETWEEN DEPARTMENTS.

PROGRAM ACCEPTS A RELATIONSHIP CHART USING THE FOLLOWING 'RELVAL' S:
RELVAL KEY
6: ABSOLUTELY CLOSE 3: ORDINARY CLOSE
5: ESPECIALLY CLOSE 2: UNIMPORTANT
4: IMPORTANT 1: UNDESIRABLE
IT ACCEPTS DATA FOR THE LAYOUT IN THE FORM OF DEPT.# AND THE #’S OF DEPTS. ADJACENT TO IT.

PROGRAM SCORES THE LAYOUT AND RECOMMENDS IMPROVEMENTS.

HIT ’RETURN’ WHEN READY

The required data is now entered.

ENTER THE # OF DEPARTMENTS 5

ENTER RELVAL BETWEEN DEPT. 1 AND DEPT. 2 = ?2
AND DEPT. 3 = ?5
AND DEPT. 4 = ?5
AND DEPT. 5 = ?2

ENTER RELVAL BETWEEN DEPT. 2 AND DEPT. 3 = ?2
AND DEPT. 4 = ?4
AND DEPT. 5 = ?2

ENTER RELVAL BETWEEN DEPT. 3 AND DEPT. 4 = ?2
AND DEPT. 5 = ?5

ENTER RELVAL BETWEEN DEPT. 4 AND DEPT. 5 = ?2

DO YOU WANT TO MAKE ANY CHANGES( Y OR N ) Y

Corrections are made by entering the numbers of the departments which changes will be made between and the corresponding corrected RELVAL.

CHANGE RELVAL BET. DEPT. 1 AND DEPT. 2 TO 6

*** CHANGE IMPLEMENTED ***

DO YOU WANT TO MAKE ANY CHANGES( Y OR N ) N

HIT ‘RETURN’ TO CONTINUE
A brief set of program instructions is now given.

**INPUT DATA FOR LAYOUT TO BE EVALUATED**

FOR EACH DEPT., ENTER THE # OF CORRESPONDING ADJACENT DEPTS.

NOTE:

- NO. OF ADJACENT DEPTS. NOT TO EXCEED 20.
- ENTER '0' TO END SEQUENCE OF ADJACENT DEPTS.

HIT 'RETURN' TO CONTINUE

The required data is now entered.

FOR DEPT. # 1
   ADJACENT DEPT. 1 IS 5
   ADJACENT DEPT. 2 IS 0

DO YOU WANT TO MAKE CHANGES, (Y OR N)? N

Corrections are made by entering the corrected adjacent department numbers.
If the RELVAL between two departments is equal to six and they are not adjacent according to the layout REL Chart, a message informs the user of this case and the evaluation stops.

DEPT.# 1 AND DEPT.# 2 SHOULD BE ADJACENT IN ANY LAYOUT ARRANGEMENT. THE LAYOUT IN HAND DOES NOT FULFILL THIS REQUIREMENT.

If the RELVAL between two departments is equal to one and the departments are adjacent according to the layout REL Chart, a message informs the user of this case and the evaluation stops.

DEPT.# 1 AND DEPT.# 5 CANNOT BE ADJACENT IN ANY LAYOUT ARRANGEMENT. THE LAYOUT IN HAND DOES NOT FULFILL THIS REQUIREMENT.
The remaining data is now entered.

FOR DEPT. # 2
   ADJACENT DEPT. 1 IS 74
   ADJACENT DEPT. 2 IS 75
   ADJACENT DEPT. 3 IS 70

FOR DEPT. # 3
   ADJACENT DEPT. 1 IS 74
   ADJACENT DEPT. 2 IS 70

FOR DEPT. # 4
   ADJACENT DEPT. 1 IS 72
   ADJACENT DEPT. 2 IS 73
   ADJACENT DEPT. 3 IS 70

FOR DEPT. # 5
   ADJACENT DEPT. 1 IS 71
   ADJACENT DEPT. 2 IS 72
   ADJACENT DEPT. 3 IS 70

While calculating the Layout Equivalent REL Chart, the program will display the following:

YOUR COMPUTER IS CALCULATING
PLEASE STAND BY
After the period in which computation takes place, the results are displayed.

**LAYOUT EQUIVALENT RELATIONSHIP CHART**

**BETWEEN DEPT.# 1**
- AND DEPT.# 2 LAYREL= 5
- AND DEPT.# 3 LAYREL= 3
- AND DEPT.# 4 LAYREL= 4
- AND DEPT.# 5 LAYREL= 6

**BETWEEN DEPT.# 2**
- AND DEPT.# 1 LAYREL= 5
- AND DEPT.# 3 LAYREL= 5
- AND DEPT.# 4 LAYREL= 6
- AND DEPT.# 5 LAYREL= 6

**BETWEEN DEPT.# 3**
- AND DEPT.# 1 LAYREL= 3
- AND DEPT.# 2 LAYREL= 5
- AND DEPT.# 4 LAYREL= 6
- AND DEPT.# 5 LAYREL= 4

**BETWEEN DEPT.# 4**
- AND DEPT.# 1 LAYREL= 4
- AND DEPT.# 2 LAYREL= 6
- AND DEPT.# 3 LAYREL= 6
- AND DEPT.# 5 LAYREL= 5

**BETWEEN DEPT.# 5**
- AND DEPT.# 1 LAYREL= 6
- AND DEPT.# 2 LAYREL= 6
- AND DEPT.# 3 LAYREL= 4
- AND DEPT.# 4 LAYREL= 5

HIT 'RETURN' TO CONTINUE
The evaluation scoring is now displayed.

EVALUATION SCORING

FROM DEPT 1 TO DEPT 2 EVAL SCORE= 5
FROM DEPT 1 TO DEPT 3 EVAL SCORE= 3
FROM DEPT 1 TO DEPT 4 EVAL SCORE= 4
FROM DEPT 1 TO DEPT 5 EVAL SCORE= 5
FROM DEPT 2 TO DEPT 3 EVAL SCORE= 5
FROM DEPT 2 TO DEPT 4 EVAL SCORE= 5
FROM DEPT 2 TO DEPT 5 EVAL SCORE= 5
FROM DEPT 3 TO DEPT 4 EVAL SCORE= 5
FROM DEPT 3 TO DEPT 5 EVAL SCORE= 4
FROM DEPT 4 TO DEPT 5 EVAL SCORE= 5

A TOTAL EVALUATION SCORE OF 46
OUT OF A POSSIBLE MAXIMUM OF 50

WAS DETERMINED.

HIT RETURN TO CONTINUE
Guidelines for rearrangement with respect to satisfying production requirements are now displayed.

IF THE FOLLOWING DEPTS.
BECAME CLOSER ---

DEPT. 1 & DEPT. 3
THIS WOULD INCREASE THE TOTAL EVALUATION SCORE TO 48
OUT OF A POSSIBLE MAXIMUM OF 50

HIT RETURN TO CONTINUE

IF THE FOLLOWING DEPTS.
BECAME CLOSER ---

DEPT. 1 & DEPT. 4
DEPT. 3 & DEPT. 5
THIS WOULD INCREASE THE TOTAL EVALUATION SCORE TO 48
OUT OF A POSSIBLE MAXIMUM OF 50

HIT RETURN TO CONTINUE
Finally, program options are displayed.

NEXT STEP TO BE PERFORMED
1. EXIT PROGRAM
2. INPUT NEW DATA

ENTER CHOICE #
CHAPTER VII
CONCLUSIONS AND RECOMMENDATIONS

Conclusions

This research has been concerned with the development of micro-software for use in plant layout design. Five programs were developed in this study which include analysis, arrangement, and evaluation techniques.

Effective analytical techniques such as CRAFT (Computerized Relative Allocation of Facilities Technique) and CORELAP (Computerized-Relationship Planning), which were originally designed for main-frame computers, were adapted to operate on the microcomputer.

The From/To Chart and the Relationship Chart are analysis type programs. They can be used alone or as input to CRAFT and CORELAP. CRAFT is an arrangement/improvement type program which requires an initial layout. CORELAP is an arrangement/construction type program which constructs a layout arrangement from "scratch". These layout arrangements can be evaluated by using the evaluation program, EVAL.

The use of micro computers could result in quicker, cheaper, and more effective analysis of layout changes because the supporting software is available.
Recommendations

During the course of my study, I have come across certain areas which should be further investigated. The following is a summary of recommended research:

1. Apply graphical techniques using high resolution graphics to display the layout arrangement to the user.

2. Develop a program/methodology by which same techniques can be used to accommodate more input data as it becomes available.

3. Analyze the effect of different arrangement techniques for use in construction programs, such as CORELAP.

4. Develop an evaluation program which would use other criteria than relationship values, such as backtracking and safety.
LIST OF REFERENCES


