Flow, space and activity relationships I.

- Departmental Planning
- Manufacturing cells
  - Clustering algorithms for cell formation
Flow, Space and Activity Relationships

- **Flow**
  - Flow into, within and from manufacturing facility
  - Flow of materials, people, equipment, information, money, etc.

- **Space**
  - The amount of space required in the facility
  - Workstation specification, department specification and other space requirements

- **Activity relationships**
  - Activity relationship is the key input in facilities design
  - Defined by flow relationships, organizational relationships, environmental relationships, process relationships and control relationships
Departmental planning

- Production planning departments are collections of workstations to be grouped together during the facilities layout process.
- Combining workstations that perform similar functions:
  - Similar products or components
  - Similar processes
- Classification of layouts based on product volume-variety:
  - Product layout (flow shop)
  - Fixed product layout
  - Group layout
  - Process layout (job shop)
Product layout (Flow shop)

Fixed product layout

Group layout

Process layout (Job shop)
## Departmental planning

<table>
<thead>
<tr>
<th>Product</th>
<th>Layout</th>
<th>Combining workstations</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Standardized • Large stable demand</td>
<td>• <strong>Product layout</strong></td>
<td>• Combine all workstations required to produce the product</td>
</tr>
<tr>
<td>• Physically large • Awkward to move • Low sporadic demand</td>
<td>• <strong>Fixed product layout</strong></td>
<td>• Combine all workstations required to produce the product with the area required for staging the product</td>
</tr>
<tr>
<td>• Capable of being grouped into families of similar parts</td>
<td>• <strong>Group layout</strong></td>
<td>• Combine all workstations required to produce the family of products</td>
</tr>
<tr>
<td>• None of the above</td>
<td>• <strong>Process layout</strong></td>
<td>• Combine identical workstations into departments</td>
</tr>
<tr>
<td></td>
<td>(job shop)</td>
<td>• Combine similar departments</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Production volume and product variety determines type of layout.
Group Technology – Cellular Manufacturing

- Group technology (product family) departments aggregate medium volume-variety parts into families based on similar manufacturing operations and design attributes.

Similar design attributes, different manufacturing requirements

Different design attributes, similar manufacturing requirements
Group Technology – Cellular Manufacturing

- Group technology (product family) departments aggregate medium volume-variety parts into families based on similar manufacturing operations and design attributes.
Group Technology – Cellular Manufacturing

- Manufacturing cells group machines, employees, materials, tooling and material handling and storage equipment to produce families of parts.
- Manufacturing cell operation needs minimum external support.
- Often designed, controlled and operated using JIT, TQM and TEI.
- Benefits of cell manufacturing:
  - **Reduction**: inventories, space, paperwork, equipment, transportation, etc.
  - **Simplification**: communication, handling, scheduling, etc.
  - **Improvement**: productivity, flexibility, quality, customer satisfaction, etc.
## Cellular Manufacturing

### Evaluation of cell design decisions

<table>
<thead>
<tr>
<th>System structure</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Equipment and tooling investment</td>
<td>Low</td>
</tr>
<tr>
<td>Equipment relocation cost</td>
<td>Low</td>
</tr>
<tr>
<td>Inter and intra cell material handling cost</td>
<td>Low</td>
</tr>
<tr>
<td>Floor space requirement</td>
<td>Low</td>
</tr>
<tr>
<td>Extend to which parts are completed in a cell</td>
<td>High</td>
</tr>
<tr>
<td>Flexibility</td>
<td>High</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>System operation</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Equipment utilization</td>
<td>High</td>
</tr>
<tr>
<td>Work-in-process inventory</td>
<td>Low</td>
</tr>
<tr>
<td>Queue lengths at each work station</td>
<td>Low</td>
</tr>
<tr>
<td>Job throughput time</td>
<td>Short</td>
</tr>
<tr>
<td>Job lateness</td>
<td>Low</td>
</tr>
</tbody>
</table>
Before GT

Welding
Turning
Heat treat
Milling
Forming
Raw materials

After GT

Raw materials
Cell #1
Cell #2
Cell #3
HT
• Before GT
• After GT
Manufacturing cell forming

- Successful implementation requires addressing selection, cell design, cell operation and cell control issues
- Manufacturing cell forming:
  - Classification
  - Production flow analysis
  - Clustering methodologies
  - Heuristic procedures
  - Mathematical models
- Cell forming is seldom the responsibility of a facility planner
Clustering methodologies

- Group parts together so that they can be processed as a family
- Links parts and machines in machine-part matrix
Clustering methodologies

- Group parts together so that they can be processed as a family
- Links parts and machines in machine-part matrix

Machine-part matrix
Direct clustering algorithm (DCA)

1. Form machine-part matrix
2. Sum the 1s in each column & row
3. Order the rows in descending order
4. Order the columns in ascending order
5. Sort the columns (1 in the first row moves left, then in the second row, etc.)
6. Sort the rows (1 in the first column moves upward, then in the second column, etc.)
7. Form sells
Direct clustering algorithm

2. Sum the 1s in each column & row
3. Order the rows in descending order
4. Order the columns in ascending order
Direct clustering algorithm

5. Sort the columns (1 in the first row moves the column to the left, then 1 in the second row, etc.)

Figure 3.3: Ordered machine-part matrix

Figure 3.4: Column-sorted machine-part matrix.
Direct clustering algorithm

6. Sort the rows (1 in the first column moves the row upward, then 1 in the second column, etc.)
7. Form cells

- Cell #1:
  - Machines 2, 4 and 5

- Cell #2:
  - Machines 1 and 3
Direct clustering algorithm
“bottleneck machines”

<table>
<thead>
<tr>
<th>Part #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th># of 1s</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>2</td>
</tr>
</tbody>
</table>

# of 1s 3 2 3 2 2 2

Figure 3.7 Machine-part matrix.

<table>
<thead>
<tr>
<th>Part #</th>
<th>5</th>
<th>4</th>
<th>2</th>
<th>3</th>
<th>1</th>
<th># of 1s</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

# of 1s 2 2 2 2 3

Figure 3.8 Ordered machine-part matrix

- Machine 2 which is needed for parts #3 and #5 creates conflicts!
Direct clustering algorithm “bottleneck machines”

What can we do?
Direct clustering algorithm
“bottleneck machines”

Possible solutions:

- Locate the bottleneck machines close to each other:
  - in different cells
  - at the boundary between cells
- Redesign the parts
- Outsource the parts
- Duplicate machines
Direct clustering algorithm
“bottleneck machines”

- Duplicating machines

![Duplicated machine 2](image1)

![Duplicated machine 3](image2)
Direct clustering algorithm
“bottleneck machines”

<table>
<thead>
<tr>
<th>Part #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th># of 1s</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

# of 1s | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |

Figure 3.11 Machine-part matrix.

<table>
<thead>
<tr>
<th>Part #</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th># of 1s</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

# of 1s | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 |

DCA
Create only 2 cells

Place the bottleneck machines at the boundary of the cells

Duplicate the bottleneck machines

Can the problem be solved by redesigning or outsourcing?
Binary Ordering Algorithm

- Binary ordering algorithm (Rank ordering algorithm) considers the rows and columns as binary strings

Procedure:
1. Compute the decimal equivalent of the binary strings for rows
2. Reorder the rows in decreasing order of their binary value
3. Compute the decimal equivalent of the binary strings for columns
4. Reorder the columns in decreasing order of their binary value
5. If the machine-part matrix is unchanged, then stop, else repeat
Binary Ordering Algorithm

- Machine-part matrix

**TABLE 12.3** Machine–Component Matrix

<table>
<thead>
<tr>
<th>Machines</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M2</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>M4</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Binary Ordering Algorithm

1. Assign binary weights from right to left for components and calculate the decimal equivalent for each row (machine)

```
<table>
<thead>
<tr>
<th>Machines</th>
<th>2⁹</th>
<th>2⁸</th>
<th>2⁷</th>
<th>2⁶</th>
<th>2⁵</th>
<th>2⁴</th>
<th>2³</th>
<th>2²</th>
<th>2¹</th>
<th>2⁰</th>
<th>decimal equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>M₁</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1007</td>
</tr>
<tr>
<td>M₂</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>451</td>
</tr>
<tr>
<td>M₃</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>568</td>
</tr>
<tr>
<td>M₄</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>455</td>
</tr>
<tr>
<td>M₅</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1020</td>
</tr>
</tbody>
</table>
```
Binary Ordering Algorithm

2. Rank the machines according to their decimal equivalent computed in the previous step

3. Assign binary weights from bottom up for machines and calculate the decimal equivalent for each column (part)

<table>
<thead>
<tr>
<th>Machines</th>
<th>Binary weight</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$2^9$</td>
</tr>
<tr>
<td>$M_5$</td>
<td>$2^4$</td>
<td>1</td>
</tr>
<tr>
<td>$M_1$</td>
<td>$2^3$</td>
<td>1</td>
</tr>
<tr>
<td>$M_3$</td>
<td>$2^2$</td>
<td>1</td>
</tr>
<tr>
<td>$M_4$</td>
<td>$2^1$</td>
<td>1</td>
</tr>
<tr>
<td>$M_2$</td>
<td>$2^0$</td>
<td>1</td>
</tr>
<tr>
<td>Column decimal equivalent</td>
<td></td>
<td>28</td>
</tr>
</tbody>
</table>
4. Reorder the columns in decreasing order of their binary value
5. If the machine-part matrix is unchanged, then stop, else repeat
Cluster Identification algorithm

1. Select any row and cross it
2. For each crossed 1 make a vertical line
3. For each crossed 1 make a horizontal line
4. Repeat until all the 1s are crossed by a vertical line or by a horizontal line
5. Form a cell from all the machines and components which were crossed
6. Remove all the crossed elements (machines and components) and start again
Cluster Identification algorithm

1. Select any row and cross it

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M6</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M7</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
Cluster Identification algorithm

2. For each crossed 1 make a vertical line

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td></td>
<td>I</td>
<td>I</td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>M4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Cluster Identification algorithm

3. For each crossed 1 make a horizontal line

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>M4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>M6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>M7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
4. Repeat until all the 1s are crossed by a vertical line or by a horizontal line
5. Form a cell from all the machines and components which were crossed

First cell is identified!
Cell #1 will produce parts P2, P3, P5 and P8 with Machines M1, M5 and M7
Cluster Identification algorithm

6. Remove all the crossed elements (machines and components) and start again

Cell #2 will produce parts P1 and P6 with Machines M2 and M4
Cluster Identification algorithm

6. Remove all the crossed elements (machines and components) and start again

Cell #3 will produce parts P4 and P7 with Machines M3 and M6
Cluster Identification algorithm

- Three resulting cells:

```
+-------+-------+-------+-------+-------+-------+-------+
|       | P2    | P3    | P5    | P8    | P1    | P6    | P4    | P7    |
+-------+-------+-------+-------+-------+-------+-------+-------+-------+
| M1    | 1     | 1     | 1     |       |       |       |       |       |
| M5    |       |       |       | 1     |       |       |       |       |
| M7    |       | 1     | 1     | 1     |       |       |       |       |
| M2    |       |       |       |       | 1     | 1     |       |       |
| M4    |       |       |       |       |       | 1     | 1     |       |
| M3    |       |       |       |       |       |       |       | 1     |
| M6    |       |       |       |       |       |       |       |       |
+-------+-------+-------+-------+-------+-------+-------+-------+-------+
```

- Cell #1
- Cell #2
- Cell #3
Cost Analysis Algorithm

- In the real-world cases the solution will rarely be straightforward. The algorithm may determine all the parts to be produced in one cell.
- Cost Analysis Algorithm allows to:
  - determine the number of machines
  - consider the cost of subcontracting
Cost Analysis Algorithm

1. Cross a column with the highest cost
2. Make a horizontal line for each crossed 1
3. Form a group of parts which are crossed only by the horizontal lines
   ◦ For each part apply the basic Cluster Identification Algorithm
   ◦ Start always with the part of the highest cost
   ◦ If the max number of machines ends up to be higher, the part is an exception (to be subcontracted)
   ◦ See if you can continue to reach the maximum number of machines
4. Form a cell with the parts and machines which are crossed
5. Eliminate the exceptions and the pieces belonging to the created cell, form a new Machine-part matrix and start again
### Cost Analysis Algorithm

1. Cross a column with the highest cost

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
<th>P11</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M6</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Cost | 2.5 | 8.0 | 70.0 | 6.0 | 15.0 | 5.0 | 10.0 | 7.0 | 2.0 | 30.0 | 4.0 |

Cells can have a **maximum of 4 machines**
## Cost Analysis Algorithm

2. Make a horizontal line for each crossed 1

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
<th>P11</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M6</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M7</td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td></td>
</tr>
<tr>
<td><strong>Cost</strong></td>
<td>2.5</td>
<td>8.0</td>
<td>70.0</td>
<td>6.0</td>
<td>15.0</td>
<td>5.0</td>
<td>10.0</td>
<td>7.0</td>
<td>2.0</td>
<td>30.0</td>
<td>4.0</td>
</tr>
</tbody>
</table>
Cost Analysis Algorithm

3. Form a group of parts which are crossed only by the horizontal lines and for each part apply the basic algorithm

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
<th>P11</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 1</td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M6</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 1</td>
<td></td>
</tr>
</tbody>
</table>

Cost: 2.5 8.0 70.0 6.0 15.0 5.0 10.0 7.0 2.0 30.0 4.0
Cost Analysis Algorithm

3. Apply the basic algorithm for part P7 – **P7 is ACCEPTED**

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
<th>P11</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M4</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M6</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M7</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Cost:**
- Cost 2.5 8.0 70.0 6.0 15.0 5.0 10.0 7.0 2.0 30.0 4.0
3. Apply the basic algorithm for part P2 – **P2 is ACCEPTED**
3. Apply the basic algorithm for part P4 – more than 4 machines would be necessary - **P4 is an EXCEPTION**

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
<th>P11</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M4</td>
<td>I</td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td>1</td>
<td>I</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>M6</td>
<td></td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td>I</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M7</td>
<td></td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td>I</td>
<td></td>
</tr>
</tbody>
</table>

| Cost | 2.5 | 8.0 | 70.0 | 6.0 | 15.0 | 5.0 | 10.0 | 7.0 | 2.0 | 30.0 | 4.0 |
3. Apply the basic algorithm for part P6 – **P6 is ACCEPTED**

### Cost Analysis Algorithm

<table>
<thead>
<tr>
<th>M1</th>
<th>M2</th>
<th>M3</th>
<th>M4</th>
<th>M5</th>
<th>M6</th>
<th>M7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
<th>P11</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5</td>
<td>8.0</td>
<td>70.0</td>
<td>6.0</td>
<td>15.0</td>
<td>5.0</td>
<td>10.0</td>
<td>7.0</td>
<td>2.0</td>
<td>30.0</td>
<td>4.0</td>
</tr>
</tbody>
</table>
Cost Analysis Algorithm

3. Apply the basic algorithm for part P1 – more than 4 machines would be necessary - **P1 is an EXCEPTION**

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
<th>P11</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>I</td>
<td>I</td>
<td></td>
<td>1</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M6</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M7</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td>I</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cost</td>
<td>2.5</td>
<td>8.0</td>
<td>70.0</td>
<td>6.0</td>
<td>15.0</td>
<td>5.0</td>
<td>10.0</td>
<td>7.0</td>
<td>2.0</td>
<td>30.0</td>
<td>4.0</td>
</tr>
</tbody>
</table>
Cost Analysis Algorithm

3. Apply the basic algorithm for part P9 – more than 4 machines would be necessary – **P9 is an EXCEPTION**
Cost Analysis Algorithm

4. Form a cell with the parts and machines which are crossed

<table>
<thead>
<tr>
<th></th>
<th>P2</th>
<th>P3</th>
<th></th>
<th>P6</th>
<th>P7</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M7</td>
<td>1</td>
<td>1</td>
<td>1 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Cell #1 will produce parts P2, P3, P6 and P7 with Machines M1, M4 and M7
5. Eliminate the exceptions and the pieces belonging to the created cell, form a new Machine-part matrix and start again.

<table>
<thead>
<tr>
<th></th>
<th>P5</th>
<th>P8</th>
<th>P10</th>
<th>P11</th>
</tr>
</thead>
<tbody>
<tr>
<td>M2</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M5</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M6</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Cost: 15 7 30 4

Cell #2 will produce parts P5, P8, P10 and P11 with Machines M2, M3, M5 and M6.
Analysis of the exceptions: Parts P4, P1 and P9 – as discussed previously

**Cost Analysis Algorithm**

- **Two resulting cells:**

<table>
<thead>
<tr>
<th></th>
<th>P2</th>
<th>P3</th>
<th>P7</th>
<th>P6</th>
<th>P11</th>
<th>P5</th>
<th>P8</th>
<th>P10</th>
<th>P1</th>
<th>P4</th>
<th>P9</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M7</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Cost**

- 2.5
- 8.0
- 70.0
- 6.0
- 15.0
- 5.0
- 10.0
- 7.0
- 2.0
- 30.0
- 4.0

Exceptions
Next lecture

• Quiz
Next lecture

- Activity relationships
- Flow patterns
- Flow planning
- Measuring the flow
- Space requirements