

P08

# How Do I Group Them?

E212 - Facilities Planning and Design



SCHOOL OF  
ENGINEERING

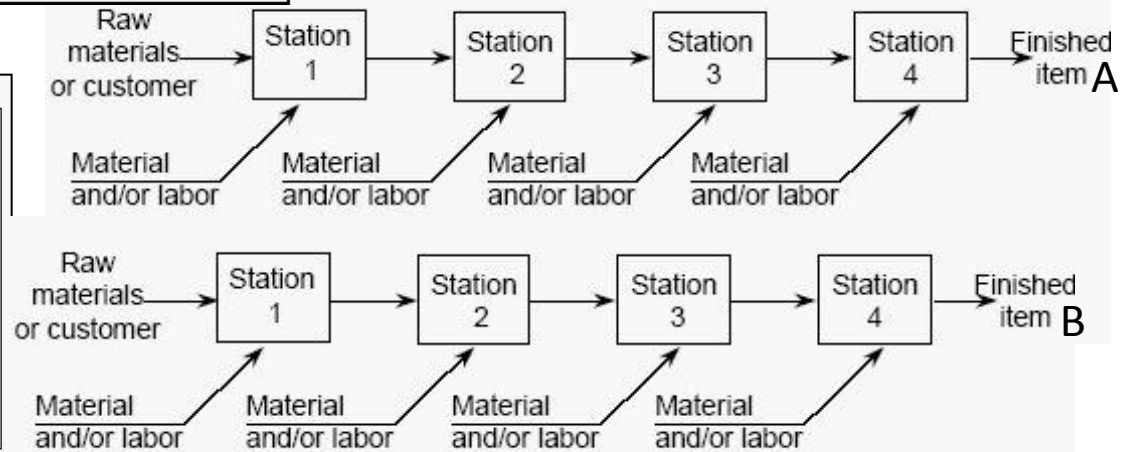
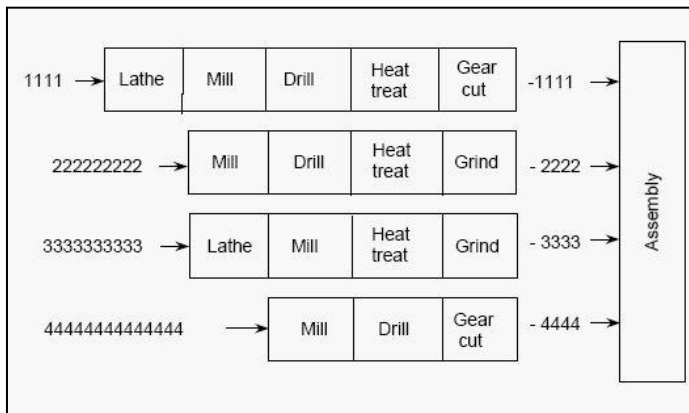
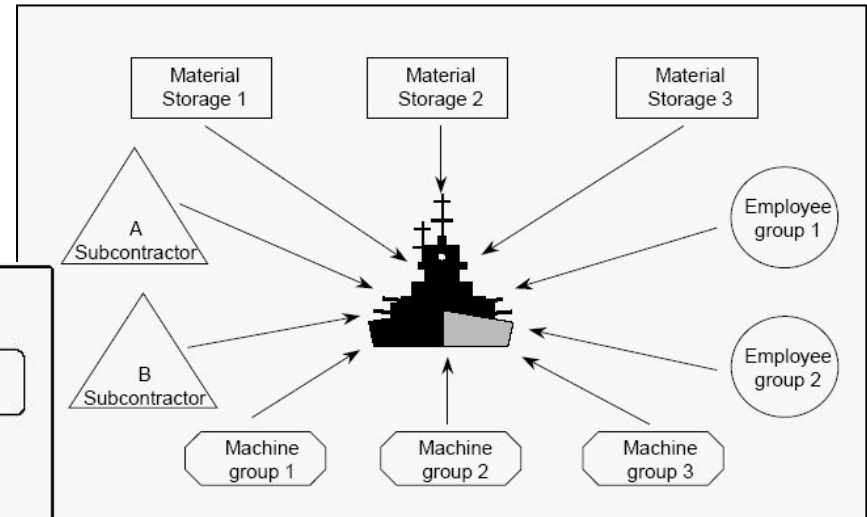
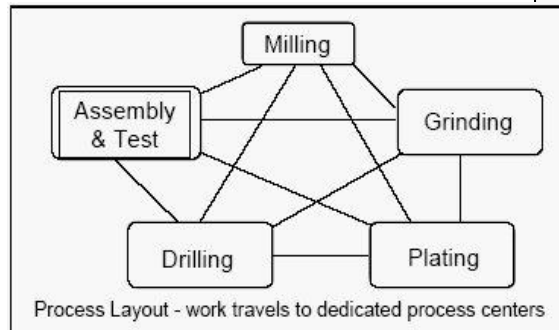
# Recap: Basic Layout Types



(Problem #03)

## Basic layout types:

1. Fixed-position
2. Product
3. Process
4. Cellular
5. Mixed



# Group Layout

---

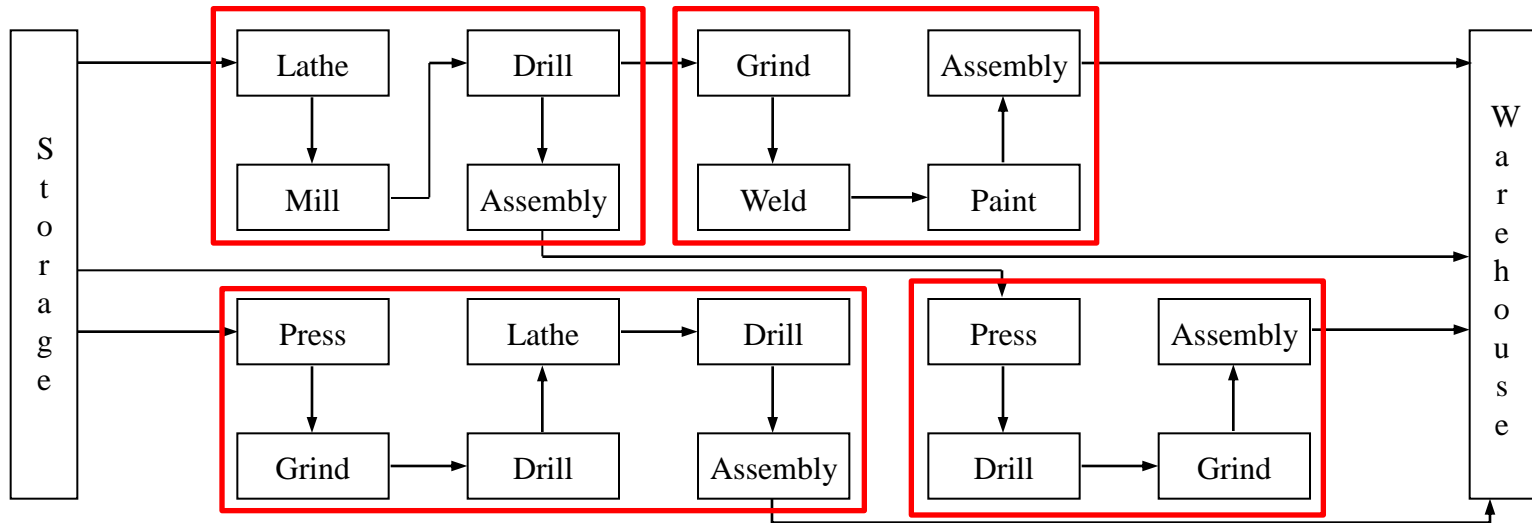


Group Layout (also known as **cellular layout** or **group technology layout**) is a type of layout in which machines are grouped into a cell that can handle items with similar processing requirements. It is often regarded as an equipment/machine layout configured to support **Cellular Manufacturing (CM)**.

In Group Layout:

- Products/parts are being grouped into “**product or part families**” based on their similarities such as common processing sequences, material composition, tooling requirements, handling requirements, storage requirements, control requirements, etc.
- The processing equipment or machines required are then grouped and arranged into a **manufacturing cell** or **work cell**, based on the corresponding product or part family that they handle.

# Group Layout



When the production volumes for the individual products are not sufficient to justify for pure product layouts, Group Layout is a better alternative as the equipment/machines can be better utilized for manufacturing the several type of products belonging to the same product family.

# Group Layout

---



Some **advantages** of Group Layout include:

1. More products (with similarities) can be produced by the same machines
2. Supports the use of general purpose machines/equipment.
3. Compromise between product layout and process layout – with associated advantages.
4. Shorter travel distances and smoother flow lines than for process layout.
5. Team attitude and job enlargement tend to occur.

Some **limitations** of Group Layout include:

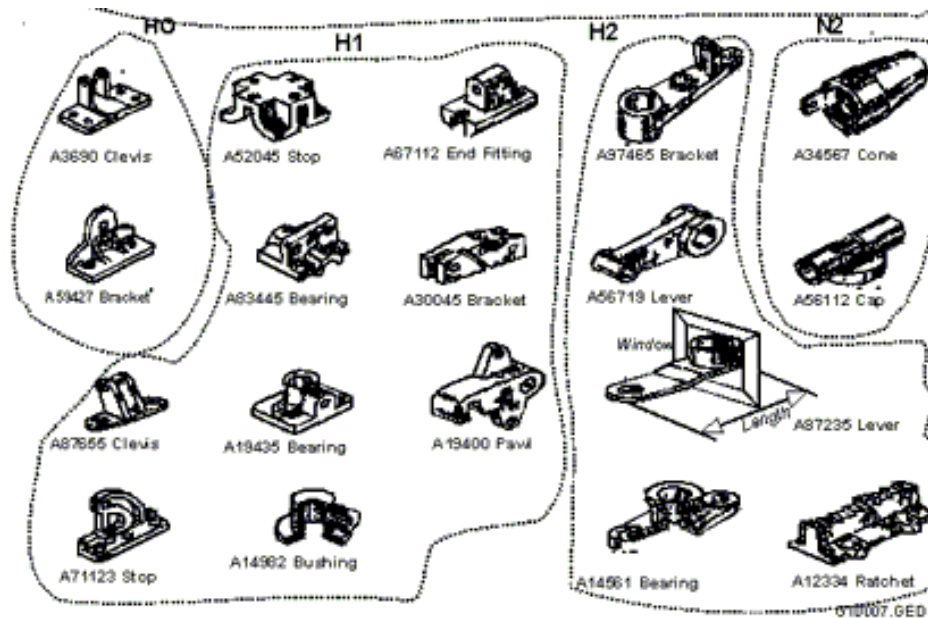
1. Higher skill levels is required of employees (compared to product layout).
2. General supervision is required.
3. Compromise between product layout and process layout – with associated limitations.
4. Depends on balanced material flow through the cell. Otherwise, buffers and work-in-process inventories will be built up.

# Group Technology (GT)



Group Technology (GT) - a management philosophy that attempts to group products / parts with similar design or manufacturing characteristics, or both.

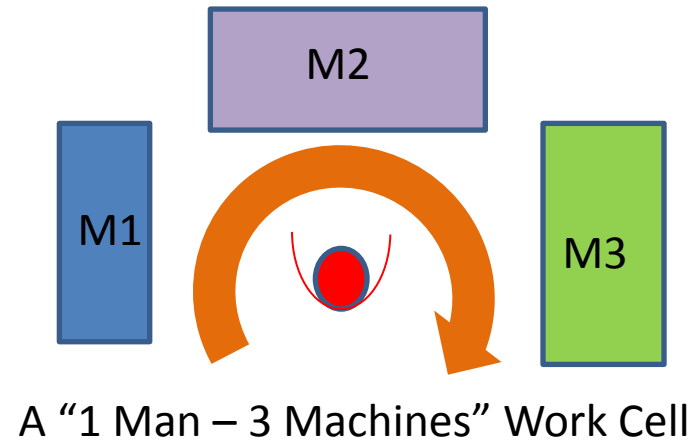
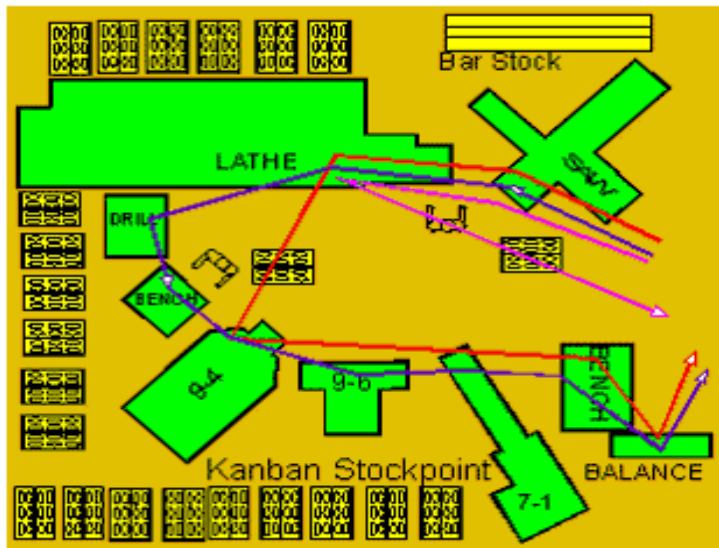
By grouping similar parts together, a common set of strategy can be developed to handle their required processing, thus saving time and effort.



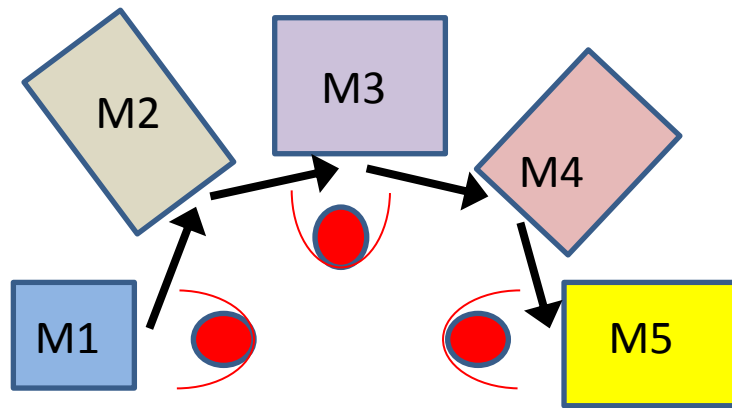
A group of similar parts is known as a “**part family**”.



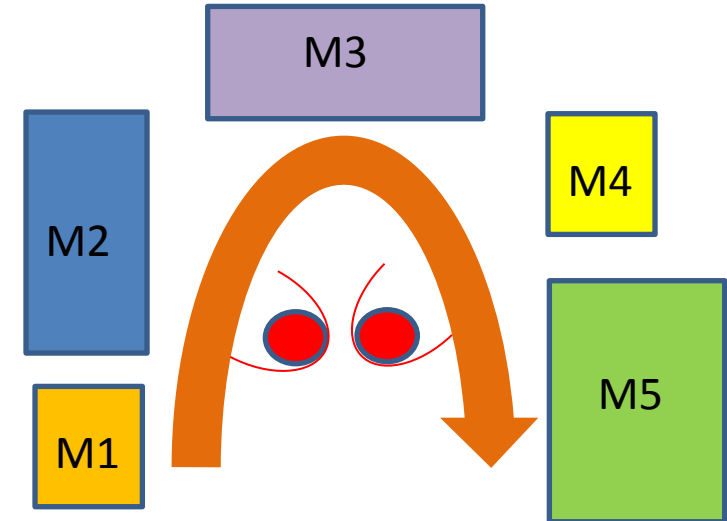
# Cellular Manufacturing (Examples)



A "1 Man – 3 Machines" Work Cell

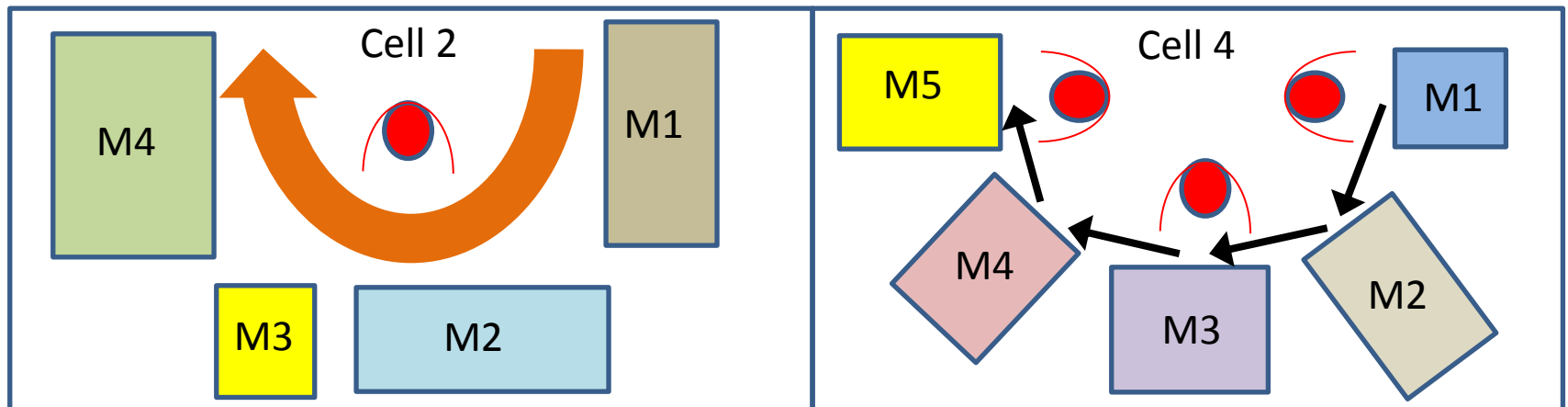
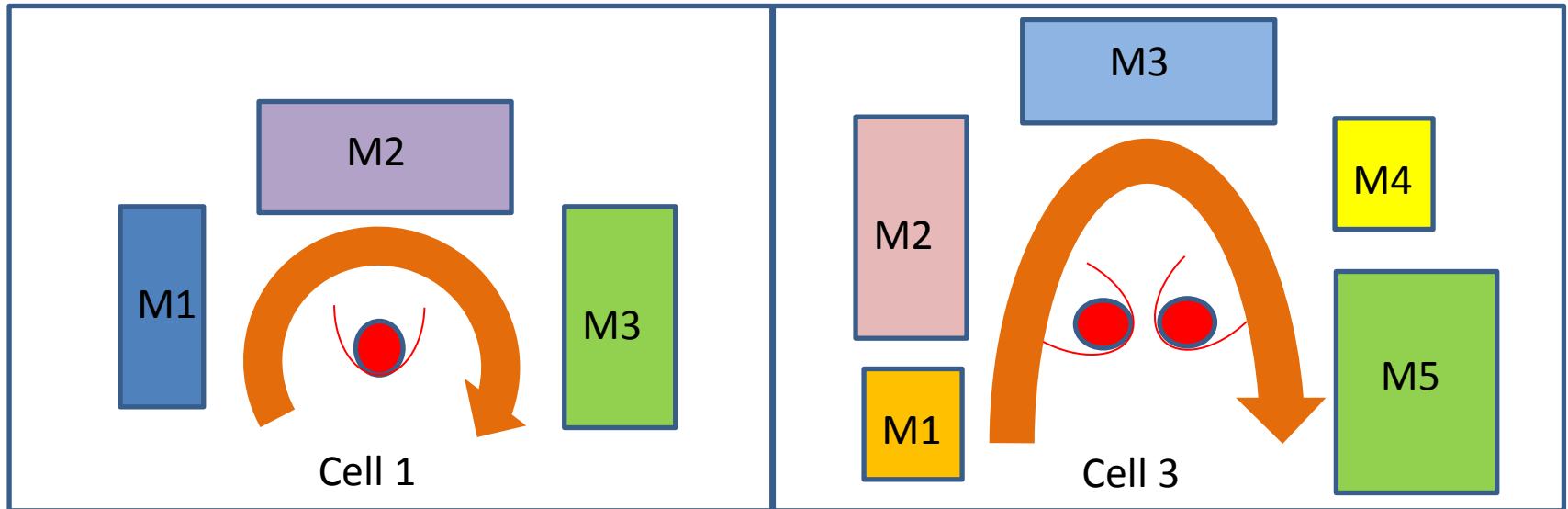


A 3 Man – 5 Machines Work Cell



A "2 Man – 5 Machines" Work Cell

# Cellular Manufacturing (Examples)



# Potential Benefits of Group Technology & Cellular Manufacturing



## **Increases:**

- Productivity. (compared to process layout)
- Components standardization.
- Reliability of estimates/forecasts.
- Costing accuracy.
- Material flow.
- Machine utilization.
- Space utilization.
- Quality.
- Customer service.
- Order potential.
- Employee morale.
- Etc...

## **Reduces:**

- Planning effort.
- Paper work
- Setup time.
- Down-time.
- Work-in-process (WIP) inventory.
- Work movement.
- Overall production time.
- Material handling cost.
- Direct/indirect labour cost.
- Overall costs.
- Etc...

# Group Technology: Clustering Techniques

---



**Clustering Techniques** - a class of methods concerned with identifying machine cells, corresponding part families, or both by attempting to rearrange the row and columns of the machine-part indicator matrix until a block diagonal form can be identified.

Some of the commonly used **clustering algorithms** include:

1. Row & Column Masking (R&CM)
2. Rank Order Clustering (ROC)

# Row & Column Masking Algorithm



The [steps](#) in using the [Row & Column Masking Algorithm](#) are as follow:

1. Draw a horizontal line across the 1st row. Select any '1' entry in the matrix which is cut through only by one line.
2. If the entry is cut through by a horizontal line, go to step 2a. If the entry is cut through by a vertical line, go to step 2b.
  - 2a. Draw a vertical line down the column in which this '1' entry appears. Go to step 3.
  - 2b. Draw a horizontal line across the row in which this '1' entry appears. Go to step 3.
3. Look for any '1' entry with only one line cutting through it, select any one and go to step 2. Repeat until there are no more such entries left. Identify the corresponding machine cell and part family. Go to step 4.
4. Select any row through which there is no line yet. If there are no such rows, stop. Otherwise, draw a horizontal line across this row, and select any '1' entry in the matrix which is cut through by only one line. Go to step 2.

Sample Solution:

---



# Using Row & Column Masking (R&CM) Algorithm

# Row & Column Masking Algorithm



Let's assign the codes below to represent all the Parts and Machines:

<b><i>Pump Parts:</i></b>	<b><i>Machines:</i></b>
P1: Pocket	M1: Sawing Machine
P2: Upper Swage	M2: Turning Machine
P3: Lower Swage	M3: Milling Machine
P4: Upper Discriminator	M4: Horizontal Boring Machine
P5: Lower Discriminator	M5: Drilling Machine
P6: Body Pipe	M6: Honing Machine
	M7: Deburring Machine

# Row & Column Masking Algorithm



		(Types of Products)					
		P1	P2	P3	P4	P5	P6
(Types of Machines/Equipment)	M1	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
	M2	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>0</b>
	M3	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>0</b>
	M4	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>
	M5	<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>
	M6	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>
	M7	<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>0</b>

**To begin:** Arrange the machines and products into a matrix as shown on the left.

Enter a '**1**' for a product (P) that is processed by a machine (M).

Enter a '**0**' for a product (P) that is not processed by a machine (M).

# Row & Column Masking Algorithm



	P1	P2	P3	P4	P5	P6
1 M1	1	0	0	0	0	0
M2	0	1	1	1	0	0
M3	0	0	1	1	1	0
M4	1	0	0	0	0	1
M5	0	1	1	0	0	0
M6	1	0	0	0	0	1
M7	0	1	0	0	1	0

**Step 1:** Draw a single horizontal line (#1) across the 1st row.

# Row & Column Masking Algorithm



		P1	P2	P3	P4	P5	P6
M1	1	1	0	0	0	0	0
M2		0	1	1	1	0	0
M3		0	0	1	1	1	0
M4		1	0	0	0	0	1
M5		0	1	1	0	0	0
M6		1	0	0	0	0	1
M7		0	1	0	0	1	0

**Step 2:** Identify the '1' entry in the matrix which is cut through only by one line. As this '1' entry is cut through by a horizontal line, draw a single vertical line down the column in which this '1' entry appear.

# Row & Column Masking Algorithm



		2					
		P1	P2	P3	P4	P5	P6
1	M1	1	0	0	0	0	0
	M2	0	1	1	1	0	0
	M3	0	0	1	1	1	0
3	M4	1	0	0	0	0	1
	M5	0	1	1	0	0	0
4	M6	1	0	0	0	0	1
	M7	0	1	0	0	1	0

**Step 3:** Identify the all the '1' entries in the matrix which is cut through only by one line. As these '1' entries are cut through by vertical lines, draw a single horizontal line across the rows in which these '1' entries appears.

# Row & Column Masking Algorithm



		2					5
		P1	P2	P3	P4	P5	P6
1	M1	1	0	0	0	0	0
	M2	0	1	1	1	0	0
	M3	0	0	1	1	1	0
3	M4	1	0	0	0	0	1
	M5	0	1	1	0	0	0
4	M6	1	0	0	0	0	1
	M7	0	1	0	0	1	0

**Step 3:** Identify all the '1' entries in the matrix which are cut through only by one line. As these '1' entries are cut through by horizontal lines, draw a single vertical line down the column in which these '1' entry appears.

# Row & Column Masking Algorithm



		2					5
		P1	P2	P3	P4	P5	P6
1	M1	1	0	0	0	0	0
	M2	0	1	1	1	0	0
	M3	0	0	1	1	1	0
3	M4	1	0	0	0	0	1
	M5	0	1	1	0	0	0
4	M6	1	0	0	0	0	1
	M7	0	1	0	0	1	0

**Step 4:** Now that it is observed that there are no more '1' entries with only a one line cutting through it, we can hence determine the first group of machine cell and part family:

**Group 1:**  
P1, P6  
M1, M4, M6

# Row & Column Masking Algorithm



		2					5
		P1	P2	P3	P4	P5	P6
1	M1	1	0	0	0	0	0
1	M2	0	1	1	1	0	0
	M3	0	0	1	1	1	0
3	M4	1	0	0	0	0	1
	M5	0	1	1	0	0	0
4	M6	1	0	0	0	0	1
	M7	0	1	0	0	1	0

**Step 5:** Select any row through which there is no line yet, and draw a double horizontal line across that row.

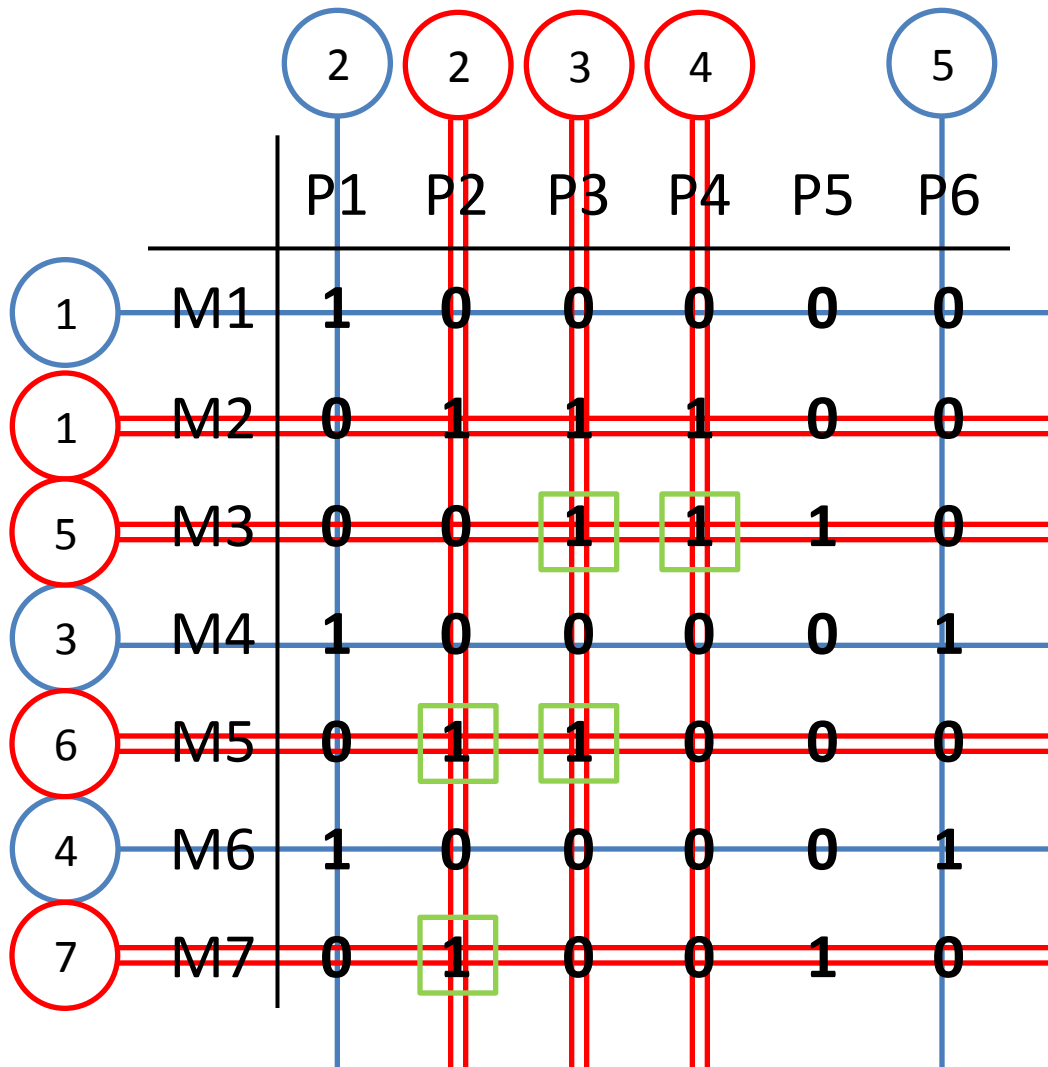
# Row & Column Masking Algorithm



		2	2	3	4	5	
		P1	P2	P3	P4	P5	P6
1	M1	1	0	0	0	0	0
1	M2	0	1	1	1	0	0
	M3	0	0	1	1	1	0
3	M4	1	0	0	0	0	1
	M5	0	1	1	0	0	0
4	M6	1	0	0	0	0	1
	M7	0	1	0	0	1	0

**Step 6:** Identify all the '1' entries in the matrix which are cut through only by one line. As these '1' entries are cut through by a horizontal line, draw a double vertical line down each of the columns in which these '1' entry appears.

# Row & Column Masking Algorithm



**Step 7:** Identify all the '1' entries in the matrix which are cut through only by one line. As these '1' entries are cut through by vertical lines, draw a double horizontal line across each of the rows in which these '1' entry appears.

# Row & Column Masking Algorithm



		2	2	3	4	8	5
		P1	P2	P3	P4	P5	P6
1	M1	1	0	0	0	0	0
1	M2	0	1	1	1	0	0
5	M3	0	0	1	1	1	0
3	M4	1	0	0	0	0	1
6	M5	0	1	1	0	0	0
4	M6	1	0	0	0	0	1
7	M7	0	1	0	0	1	0

**Step 8:** Identify all the '1' entries in the matrix which are cut through only by one line. As these '1' entries are cut through by horizontal lines, draw a double vertical line down the column in which these '1' entry appears.

# Row & Column Masking Algorithm



		2	2	3	4	8	5
		P1	P2	P3	P4	P5	P6
1	M1	1	0	0	0	0	0
1	M2	0	1	1	1	0	0
5	M3	0	0	1	1	1	0
3	M4	1	0	0	0	0	1
6	M5	0	1	1	0	0	0
4	M6	1	0	0	0	0	1
7	M7	0	1	0	0	1	0

**Step 9:** Now that it is observed that there are no more '1' entries with only a one line cutting through it, we can hence determine the second group of machine cell and part family:

**Group 2:**  
 P2, P3, P4, P5  
 M2, M3, M5, M7

# Row & Column Masking Algorithm



		2	2	3	4	8	5
		P1	P2	P3	P4	P5	P6
1	M1	1	0	0	0	0	0
1	M2	0	1	1	1	0	0
5	M3	0	0	1	1	1	0
3	M4	1	0	0	0	0	1
6	M5	0	1	1	0	0	0
4	M6	1	0	0	0	0	1
7	M7	0	1	0	0	1	0

**Conclusion:** The 2 groups of machine cells and their corresponding part families are summarized below.

**Group 1:**  
P1, P6  
M1, M4, M6

**Group 2:**  
P2, P3, P4, P5  
M2, M3, M5, M7

# Row & Column Masking Algorithm



The following grouping, as derived from the [Row & Column Masking Algorithm](#), shall be presented to Dominic for consideration:

Group 1		Group 2	
Product Family 1	Machine Cell 1	Product Family 2	Machine Cell 2
<b>P1:</b> Pocket	<b>M1:</b> Sawing Machine	<b>P2:</b> Upper Swage	<b>M2:</b> Turning Machine
<b>P6:</b> Body Pipe	<b>M4:</b> Horizontal Boring Machine	<b>P3:</b> Lower Swage	<b>M3:</b> Milling Machine
	<b>M6:</b> Honing Machine	<b>P4:</b> Upper Discriminator	<b>M5:</b> Drilling Machine
		<b>P5:</b> Lower Discriminator	<b>M7:</b> Deburring Machine

# Possible Limitation

---



- If the machine-part matrix contains one or more machines that belong to more than one cell, or contains parts that are processed in more than one cell, using R&CM may provide a solution with all machines in a cell and all parts in a corresponding part family – unable to differentiate!

# Rank Order Clustering Algorithm



The [steps](#) in using the [Rank Order Clustering Algorithm](#) are as follow:

1. Assign **binary weights**  $BW_i = 2^{m-i}$  to each **row**  $i$  of the machine-part indicator matrix, where  $m$  is the number of machines and  $n$  is the number of parts.
2. Determine the **decimal equivalent (DE)** of the binary value of each **column**  $j$  using the formula:
3. Rank the columns in decreasing order of their DE values. Break ties arbitrarily.. Rearrange the columns based on this ranking. If no rearrangement is necessary, stop. Otherwise, go to step 4.
4. For each rearranged **column**  $j$  of the matrix, assign **binary weights**  $BW_j = 2^{n-j}$ .
5. Determine the **decimal equivalent (DE)** of the binary value of each **row**  $i$  using the formula:
6. Rank the rows in decreasing order of their DE values. Break ties arbitrarily. Rearrange the rows based on this ranking. If no rearrangement is necessary, stop. Otherwise, go to step 1.

Sample Solution:

---



# Using Rank Order Clustering (ROC) Algorithm

# Rank Order Clustering(ROC) Algorithm



Let's assign the codes below to represent all the Parts and Machines:

<b><i>Pump Parts:</i></b>	<b><i>Machines:</i></b>
P1: Pocket	M1: Sawing Machine
P2: Upper Swage	M2: Turning Machine
P3: Lower Swage	M3: Milling Machine
P4: Upper Discriminator	M4: Horizontal Boring Machine
P5: Lower Discriminator	M5: Drilling Machine
P6: Body Pipe	M6: Honing Machine
	M7: Deburring Machine

# Rank Order Clustering(ROC) Algorithm



		(Types of Products)					
		P1	P2	P3	P4	P5	P6
(Types of Machines/Equipment)	M1	1	0	0	0	0	0
	M2	0	1	1	1	0	0
	M3	0	0	1	1	1	0
	M4	1	0	0	0	0	1
	M5	0	1	1	0	0	0
	M6	1	0	0	0	0	1
	M7	0	1	0	0	1	0

**To begin:** Arrange the machines and products into a matrix as shown on the left.

Enter a '**1**' for a product (P) that is processed by a machine (M).

Enter a '**0**' for a product (P) that is not processed by a machine (M).

# Rank Order Clustering(ROC) Algorithm



	P1	P2	P3	P4	P5	P6	
M1	1	0	0	0	0	0	$2^6$
M2	0	1	1	1	0	0	$2^5$
M3	0	0	1	1	1	0	$2^4$
M4	1	0	0	0	0	1	$2^3$
M5	0	1	1	0	0	0	$2^2$
M6	1	0	0	0	0	1	$2^1$
M7	0	1	0	0	1	0	$2^0$

**Step 1:** Assign binary weight  $BW_i = 2^{m-i}$  to each row  $i$  of the matrix, where  $m=7$  (the number of machines).

↑  
Assign  
Binary Weights  
to each row  $i$

# Rank Order Clustering(ROC) Algorithm



	P1	P2	P3	P4	P5	P6	
M1	1	0	0	0	0	0	64
M2	0	1	1	1	0	0	32
M3	0	0	1	1	1	0	16
M4	1	0	0	0	0	1	8
M5	0	1	1	0	0	0	4
M6	1	0	0	0	0	1	2
M7	0	1	0	0	1	0	1

**Step 2:** Calculate the **decimal equivalent (DE)** of the binary values of each **column j** using the formula:

$$DE_j = \sum_{i=1}^m (BW_i)(a_{ij})$$

DE of the →  
binary values  
of each column j

74	37	52	48	17	10
----	----	----	----	----	----

e.g.  $DE_2 = 64 \times 0 + 32 \times 1 + 16 \times 0 + 8 \times 0 + 4 \times 1 + 2 \times 0 + 1 \times 1$   
 $= 0 + 32 + 0 + 0 + 4 + 0 + 1 = 37$

# Rank Order Clustering(ROC) Algorithm



	P1	P2	P3	P4	P5	P6	
M1	1	0	0	0	0	0	64
M2	0	1	1	1	0	0	32
M3	0	0	1	1	1	0	16
M4	1	0	0	0	0	1	8
M5	0	1	1	0	0	0	4
M6	1	0	0	0	0	1	2
M7	0	1	0	0	1	0	1

**Step 3:** Rank the columns in decreasing order of their DE values.

74	37	52	48	17	10
----	----	----	----	----	----

Rank of DE values →

1 4 2 3 5 6

# Rank Order Clustering(ROC) Algorithm



	P1	P3	P4	P2	P5	P6
M1	1	0	0	0	0	0
M2	0	1	1	1	0	0
M3	0	1	1	0	1	0
M4	1	0	0	0	0	1
M5	0	1	0	1	0	0
M6	1	0	0	0	0	1
M7	0	0	0	1	1	0

**Step 4:** Re-arrange the columns in the running order of the rankings.

74	52	48	37	17	10
----	----	----	----	----	----

Columns  
re-arranged → 1 2 3 4 5 6  
in order of Rankings

# Rank Order Clustering(ROC) Algorithm



	P1	P3	P4	P2	P5	P6
M1	1	0	0	0	0	0
M2	0	1	1	1	0	0
M3	0	1	1	0	1	0
M4	1	0	0	0	0	1
M5	0	1	0	1	0	0
M6	1	0	0	0	0	1
M7	0	0	0	1	1	0

**Step 5:** Assign binary weight  $BW_j = 2^{n-j}$  to each **column**  $j$  of the matrix, where  $n=6$  (the number of parts).

$2^5$     $2^4$     $2^3$     $2^2$     $2^1$     $2^0$

← Assign Binary Weights to each column  $j$

# Rank Order Clustering(ROC) Algorithm



	P1	P3	P4	P2	P5	P6	
M1	1	0	0	0	0	0	32
M2	0	1	1	1	0	0	28
M3	0	1	1	0	1	0	26
M4	1	0	0	0	0	1	33
M5	0	1	0	1	0	0	20
M6	1	0	0	0	0	1	33
M7	0	0	0	1	1	0	6
	32	16	8	4	2	1	

**Step 6:** Calculate the **decimal equivalent (DE)** of the binary values of each row  $i$  using the formula:

$$DE_i = \sum_{j=1}^m (BW_j)(a_{ij})$$

e.g.  $DE_6 = 32 \times 1 + 16 \times 0 + \dots + 1 \times 1$   
 $= 32 + 1 = 33$

DE of the binary values of each row  $i$

# Rank Order Clustering(ROC) Algorithm



	P1	P3	P4	P2	P5	P6		
M1	1	0	0	0	0	0	32	3
M2	0	1	1	1	0	0	28	4
M3	0	1	1	0	1	0	26	5
M4	1	0	0	0	0	1	33	1
M5	0	1	0	1	0	0	20	6
M6	1	0	0	0	0	1	33	2
M7	0	0	0	1	1	0	6	7
	32	16	8	4	2	1		

**Step 7:** Rank the rows in decreasing order of their DE values.

↑  
Rank of  
DE values

# Rank Order Clustering(ROC) Algorithm



	P1	P3	P4	P2	P5	P6		
M4	1	0	0	0	0	1	33	1
M6	1	0	0	0	0	1	33	2
M1	1	0	0	0	0	0	32	3
M2	0	1	1	1	0	0	28	4
M3	0	1	1	0	1	0	26	5
M5	0	1	0	1	0	0	20	6
M7	0	0	0	1	1	0	6	7

**Step 8:** Re-arrange the rows in the running order of the rankings.

↑  
Rows re-arranged  
in order of Rankings

# Rank Order Clustering(ROC) Algorithm



	P1	P3	P4	P2	P5	P6	
M4	1	0	0	0	0	1	$2^6$
M6	1	0	0	0	0	1	$2^5$
M1	1	0	0	0	0	0	$2^4$
M2	0	1	1	1	0	0	$2^3$
M3	0	1	1	0	1	0	$2^2$
M5	0	1	0	1	0	0	$2^1$
M7	0	0	0	1	1	0	$2^0$

**Step 9:** Assign binary weight  $BW_i = 2^{m-i}$  to each row  $i$  of the matrix, where  $m=7$  (the number of machines).

(Note that this is a repeat of step 1 again)

↑  
Assign  
Binary Weights  
to each row  $i$

# Rank Order Clustering(ROC) Algorithm



	P1	P3	P4	P2	P5	P6	
M4	1	0	0	0	0	1	64
M6	1	0	0	0	0	1	32
M1	1	0	0	0	0	0	16
M2	0	1	1	1	0	0	8
M3	0	1	1	0	1	0	4
M5	0	1	0	1	0	0	2
M7	0	0	0	1	1	0	1

**Step 10:** Calculate the **decimal equivalent (DE)** of the binary values of each **column  $j$**  using the formula:

$$DE_j = \sum_{i=1}^m (BW_i)(a_{ij})$$

DE of the → 112 14 12 11 5 96  
binary values  
of each column  $j$

# Rank Order Clustering(ROC) Algorithm



	P1	P3	P4	P2	P5	P6	
M4	1	0	0	0	0	1	64
M6	1	0	0	0	0	1	32
M1	1	0	0	0	0	0	16
M2	0	1	1	1	0	0	8
M3	0	1	1	0	1	0	4
M5	0	1	0	1	0	0	2
M7	0	0	0	1	1	0	1
	112	14	12	11	5	96	

**Step 11:** Rank the columns in decreasing order of their DE values.

Rank of DE values →

1    3    4    5    6    2

# Rank Order Clustering(ROC) Algorithm



	P1	P6	P3	P4	P2	P5
M4	1	1	0	0	0	0
M6	1	1	0	0	0	0
M1	1	0	0	0	0	0
M2	0	0	1	1	1	0
M3	0	0	1	1	0	1
M5	0	0	1	0	1	0
M7	0	0	0	0	1	1

**Step 12:** Re-arrange the columns in the running order of the rankings.

112	96	14	12	11	5
-----	----	----	----	----	---

Columns re-arranged in order of Rankings → 1 2 3 4 5 6

# Rank Order Clustering(ROC) Algorithm



	P1	P6	P3	P4	P2	P5
M4	1	1	0	0	0	0
M6	1	1	0	0	0	0
M1	1	0	0	0	0	0
M2	0	0	1	1	1	0
M3	0	0	1	1	0	1
M5	0	0	1	0	1	0
M7	0	0	0	0	1	1

**Step 13:** Assign binary weight  $BW_j = 2^{n-j}$  to each **column**  $j$  of the matrix, where  $n=6$  (the number of parts).

$2^5$     $2^4$     $2^3$     $2^2$     $2^1$     $2^0$

← Assign Binary Weights to each column  $j$

# Rank Order Clustering(ROC) Algorithm



	P1	P6	P3	P4	P2	P5	
M4	1	1	0	0	0	0	48
M6	1	1	0	0	0	0	48
M1	1	0	0	0	0	0	32
M2	0	0	1	1	1	0	14
M3	0	0	1	1	0	1	13
M5	0	0	1	0	1	0	10
M7	0	0	0	0	1	1	3
	32	16	8	4	2	1	

**Step 14:** Calculate the decimal equivalent (DE) of the binary values of each row  $i$  using the formula:

$$DE_i = \sum_{j=1}^m (BW_j)(a_{ij})$$

↑  
DE of the binary values  
of each row  $i$

# Rank Order Clustering(ROC) Algorithm



	P1	P6	P3	P4	P2	P5		
M4	1	1	0	0	0	0	48	1
M6	1	1	0	0	0	0	48	2
M1	1	0	0	0	0	0	32	3
M2	0	0	1	1	1	0	14	4
M3	0	0	1	1	0	1	13	5
M5	0	0	1	0	1	0	10	6
M7	0	0	0	0	1	1	3	7
	32	16	8	4	2	1		

**Step 15:** Rank the rows in decreasing order of their DE values.

Since the ranking is now neatly arranged in order, stop the process. We can now identify the groupings.

**Group 1:**  
P1, P6  
M1, M4, M6

**Group 2:**  
P2, P3, P4, P5  
M2, M3, M5, M7



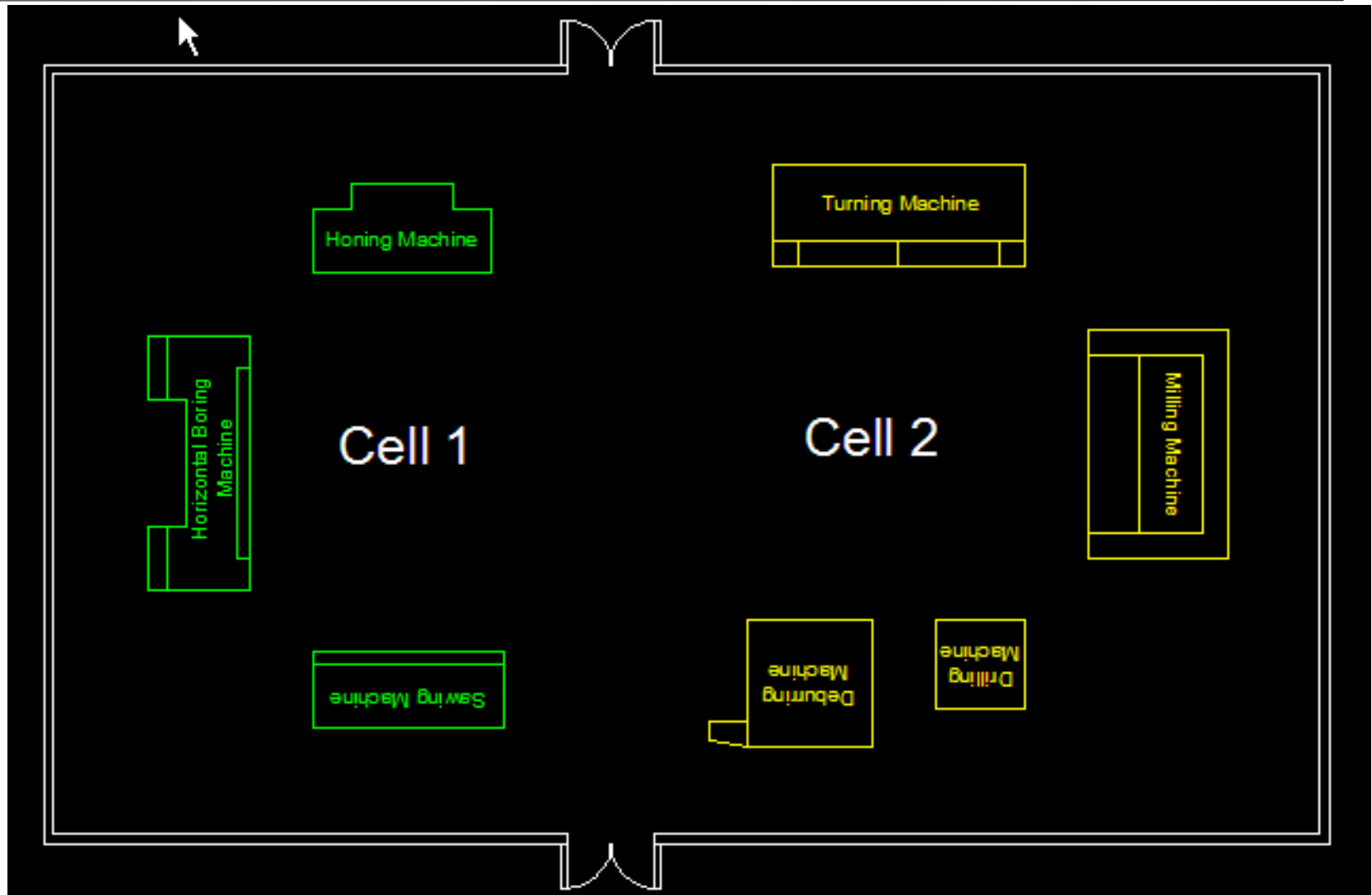
# Rank Order Clustering(ROC) Algorithm



The following grouping, as derived from the [Rank Order Clustering Algorithm](#), shall be presented to Dominic for consideration:

Group 1		Group 2	
Product Family 1	Machine Cell 1	Product Family 2	Machine Cell 2
<b>P1:</b> Pocket	<b>M1:</b> Sawing Machine	<b>P2:</b> Upper Swage	<b>M2:</b> Turning Machine
<b>P6:</b> Body Pipe	<b>M4:</b> Horizontal Boring Machine	<b>P3:</b> Lower Swage	<b>M3:</b> Milling Machine
	<b>M6:</b> Honing Machine	<b>P4:</b> Upper Discriminator	<b>M5:</b> Drilling Machine
		<b>P5:</b> Lower Discriminator	<b>M7:</b> Deburring Machine

# Suggested Layout



# Going Further...

---



- What happens if you have parts that cannot be completed in one cell?
  - Modify manufacturing process of parts (may involves modifications to parts design)
  - Design effective inter-cellular material handling system.
  - Duplicate machines.

# Learning Objectives

---



- Understand Group Layout, Group Technology, Cellular Manufacturing, and their associated advantages and limitations.
- Know how to use the Row & Column Masking (R&CM) Algorithm to group parts and machines into manufacturing cells.
- Know how to use the Rank Order Clustering (ROC) Algorithm to group parts and machines into manufacturing cells.
- Understand the similarity and differences between using Row & Column Masking Algorithm vs Rank Order Clustering Algorithm.
- Able to develop an appropriate cell manufacturing (or work cell) design.