Operation Research

Module 5

5.1 Introduction to Game Theory

Game theory is a type of decision theory in which one's choice of action is determined after taking into account all possible alternatives available to an opponent playing the same game, rather than just by the possibilities of several outcome results. Game theory does not insist on how a game should be played but tells the procedure and principles by which action should be selected. Thus it is a decision theory useful in competitive situations.

Game is defined as an activity between two or more persons according to a set of rules at the end of which each person receives some benefit or suffers loss. The set of rules defines the **game**. Going through the set of rules once by the participants defines a **play**.

5.2 Properties of a Game

1. There are finite numbers of competitors called 'players'
2. Each player has a finite number of possible courses of action called 'strategies'
3. All the strategies and their effects are known to the players but player does not know which strategy is to be chosen.
4. A game is played when each player chooses one of his strategies. The strategies are assumed to be made simultaneously with an outcome such that no player knows his opponents strategy until he decides his own strategy.
5. The game is a combination of the strategies and in certain units which determines the gain or loss.
6. The figures shown as the outcomes of strategies in a matrix form are called 'pay-off matrix'.
7. The player playing the game always tries to choose the best course of action which results in optimal pay off called 'optimal strategy'.
8. The expected pay off when all the players of the game follow their optimal strategies is known as 'value of the game'. The main objective of a problem of a game is to find the value of the game.
9. The game is said to be 'fair' game if the value of the game is zero otherwise it s known as 'unfair'.

5.3 Characteristics of Game Theory

1. Competitive game

A competitive situation is called a **competitive game** if it has the following four properties

1. There are finite number of competitors such that $n \geq 2$. In case $n = 2$, it is called a **two-person game** and in case $n > 2$, it is referred as **n-person game**.
2. Each player has a list of finite number of possible activities.
3. A play is said to occur when each player chooses one of his activities. The choices are assumed to be made simultaneously i.e. no player knows the choice of the other until he has decided on his own.

4. Every combination of activities determines an outcome which results in a gain of payments to each player, provided each player is playing uncompromisingly to get as much as possible. Negative gain implies the loss of same amount.

2. Strategy
The strategy of a player is the predetermined rule by which player decides his course of action from his own list during the game. The two types of strategy are

1. Pure strategy
2. Mixed strategy

Pure Strategy
If a player knows exactly what the other player is going to do, a deterministic situation is obtained and objective function is to maximize the gain. Therefore, the pure strategy is a decision rule always to select a particular course of action.

Mixed Strategy
If a player is guessing as to which activity is to be selected by the other on any particular occasion, a probabilistic situation is obtained and objective function is to maximize the expected gain. Thus the mixed strategy is a selection among pure strategies with fixed probabilities.

3. Number of persons
A game is called ‘n’ person game if the number of persons playing is ‘n’. The person means an individual or a group aiming at a particular objective.

Two-person, zero-sum game
A game with only two players (player A and player B) is called a ‘two-person, zero-sum game’, if the losses of one player are equivalent to the gains of the other so that the sum of their net gains is zero.
Two-person, zero-sum games are also called rectangular games as these are usually represented by a payoff matrix in a rectangular form.

4. Number of activities
The activities may be finite or infinite.

5. Payoff
The quantitative measure of satisfaction a person gets at the end of each play is called a payoff

6. Payoff matrix
Suppose the player A has ‘m’ activities and the player B has ‘n’ activities. Then a payoff matrix can be formed by adopting the following rules

- Row designations for each matrix are the activities available to player A
- Column designations for each matrix are the activities available to player B
- Cell entry V_{ij} is the payment to player A in A’s payoff matrix when A chooses the activity i and B chooses the activity j.

- With a zero-sum, two-person game, the cell entry in the player B's payoff matrix will be negative of the corresponding cell entry V_{ij} in the player A's payoff matrix so that sum of payoff matrices for player A and player B is ultimately zero.

7. Value of the game
Value of the game is the maximum guaranteed game to player A (maximizing player) if both the players uses their best strategies. It is generally denoted by 'V' and it is unique.

5.4 Classification of Games

All games are classified into
- Pure strategy games
- Mixed strategy games

The method for solving these two types varies. By solving a game, we need to find best strategies for both the players and also to find the value of the game.

Pure strategy games can be solved by **saddle point method.**

The different methods for solving a mixed strategy game are
- Analytical method
- Graphical method
- Dominance rule
- Simplex method

Solving Two-Person and Zero-Sum Game

Two-person zero-sum games may be deterministic or probabilistic. The deterministic games will have saddle points and pure strategies exist in such games. In contrast, the probabilistic games will have no saddle points and mixed strategies are taken with the help of probabilities.

Definition of saddle point
A saddle point of a matrix is the position of such an element in the payoff matrix, which is minimum in its row and the maximum in its column.

Procedure to find the saddle point
- Select the minimum element of each row of the payoff matrix and mark them with circles.
- Select the maximum element of each column of the payoff matrix and mark them with squares.
- If their appears an element in the payoff matrix with a circle and a square together then that position is called saddle point and the element is the value of the game.

Solution of games with saddle point
To obtain a solution of a game with a saddle point, it is feasible to find out

- Best strategy for player A
- Best strategy for player B
- The value of the game

The best strategies for player A and B will be those which correspond to the row and column respectively through the saddle point.

Examples

Solve the payoff matrix

1.

Player B

		I	II	III	IV	V
	I II	-2	0	0	5	3
Player A	III	3	2	1	2	2
	IV	-4	-3	0	-2	6
		5	3	-4	2	-6

Solution

Player B

		I	II	III	IV	V	
	I	-2	0	0	$\boxed{5}$	3	-2
Player A	II	3	2	$\boxed{1}$	2	2	1 Maximin value
	III	-4	-3	0	-2	$\boxed{6}$	-4
	IV	$\boxed{5}$	$\boxed{3}$	-4	2	-6	-6
		5	3	1	5	6	

Minimax value

Strategy of player A – II
Strategy of player B - III
Value of the game = 1

2.

	B1	B2	B3	B4
A1	1	7	3	4
A2	5	6	4	5

A3	7	2	0	3

Solution

	B1	B2	B3	B4	
A1	1	$\boxed{7}$	3	4	1
A2	5	6	$\boxed{4}$	$\boxed{5}$	4 Maximin value
A3	$\boxed{7}$	2	0	3	0
	7	7	4	5	

Minimax value

Strategy of player A – A2
Strategy of player B – B3
Value of the game = 4

3.

		B's Strategy				
		B1	B2	B3	B4	B5
A's Strategy	A1	8	10	-3	-8	-12
	A2	3	6	0	6	12
	A3	7	5	-2	-8	17
	A4	-11	12	-10	10	20
	A5	-7	0	0	6	2

Solution

B's Strategy

A's Strategy	B1	B2	B3	B4	B5	
A1	$\boxed{8}$	10	-3	-8	-12	-12
A2	3	6	$\boxed{0}$	6	12	0 Maximin value
A3	7	5	-2	-8	17	-8
A4	-11	$\boxed{12}$	-10	$\boxed{10}$	$\boxed{20}$	-11
A5	-7	0	$\boxed{0}$	6	2	-7
	8	12	0	10	20	

Minimax value

Strategy of player A – A2
Strategy of player B – B3
Value of the game = 0

4.

$$\begin{bmatrix} 9 & 3 & 1 & 8 & 0 \\ 6 & 5 & 4 & 6 & 7 \\ 2 & 4 & 3 & 3 & 8 \\ 5 & 6 & 2 & 2 & 1 \end{bmatrix}$$

Solution

$\boxed{9}$	3	1	$\boxed{8}$	0	0
6	5	$\boxed{4}$	6	7	4 Maximin value
2	4	3	3	$\boxed{8}$	2
5	$\boxed{6}$	2	2	1	1
9	6	4	8	8	

Minimax value

Value of the game = 4

Games with Mixed Strategies

In certain cases, no pure strategy solutions exist for the game. In other words, saddle point does not exist. In all such game, both players may adopt an optimal blend of the strategies called **Mixed Strategy** to find a saddle point. The optimal mix for each player may be determined by assigning each strategy a probability of it being chosen. Thus these mixed strategies are probabilistic combinations of available better strategies and these games hence called **Probabilistic games**.

The probabilistic mixed strategy games without saddle points are commonly solved by any of the following methods

Sl. No.	**Method**	**Applicable to**
1	Analytical Method	2x2 games
2	Graphical Method	2x2, mx2 and 2xn games
3	Simplex Method	2x2, mx2, 2xn and mxn games

3.1.1 Analytical Method

A 2 x 2 payoff matrix where there is no saddle point can be solved by analytical method.
Given the matrix

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

Value of the game is

$$V = \frac{a_{11}a_{22} - a_{21}a_{12}}{(a_{11}+a_{22})-(a_{12}+a_{21})}$$

With the coordinates

$$x_1 = \frac{a_{22}-a_{21}}{(a_{11}+a_{22})-(a_{12}+a_{21})}, \quad x_2 = \frac{a_{11}-a_{12}}{(a_{11}+a_{22})-(a_{12}+a_{21})}$$

$$y_1 = \frac{a_{22}-a_{12}}{(a_{11}+a_{22})-(a_{12}+a_{21})}, \quad y_2 = \frac{a_{11}-a_{21}}{(a_{11}+a_{22})-(a_{12}+a_{21})}$$

Alternative procedure to solve the strategy
- Find the difference of two numbers in column 1 and enter the resultant under column 2. Neglect the negative sign if it occurs.
- Find the difference of two numbers in column 2 and enter the resultant under column 1. Neglect the negative sign if it occurs.
- Repeat the same procedure for the two rows.

1. Solve

$$\begin{array}{cc} & B \\ A & \begin{bmatrix} 5 & 1 \\ 3 & 4 \end{bmatrix} \end{array}$$

Solution

It is a 2 x 2 matrix and no saddle point exists. We can solve by analytical method

$$\begin{array}{ccc} & B & \\ A & \begin{bmatrix} 5 & 1 \\ 3 & 4 \end{bmatrix} & \begin{matrix} 1 \\ 4 \end{matrix} \\ & \begin{matrix} 3 & \quad 2 \end{matrix} & \end{array}$$

$$V = \frac{a_{11}a_{22} - a_{21}a_{12}}{(a_{11}+a_{22})-(a_{12}+a_{21})} = \frac{20-3}{9-4}$$

V = 17 / 5

$S_A = (x_1, x_2) = (1/5, 4/5)$

$S_B = (y_1, y_2) = (3/5, 2/5)$

2. Solve the given matrix

$$\begin{array}{cc} & B \\ A & \begin{bmatrix} 2 & -1 \\ -1 & 0 \end{bmatrix} \end{array}$$

Solution

$$\begin{array}{ccc} & B & \\ A & \begin{bmatrix} 2 & -1 \\ -1 & 0 \end{bmatrix} & \begin{array}{c} 1 \\ 3 \end{array} \\ & \begin{array}{cc} 1 & 3 \end{array} & \end{array}$$

$$V = \frac{a_{11} a_{22} - a_{21} a_{12}}{(a_{11} + a_{22}) - (a_{12} + a_{21})} = \frac{0 - 1}{2 + 2}$$

V = - 1 / 4

$S_A = (x_1, x_2) = (1/4, 3/4)$

$S_B = (y_1, y_2) = (1/4, 3/4)$

Graphical method

The graphical method is used to solve the games whose payoff matrix has

- Two rows and n columns (2 x n)
- m rows and two columns (m x 2)

Algorithm for solving 2 x n matrix games

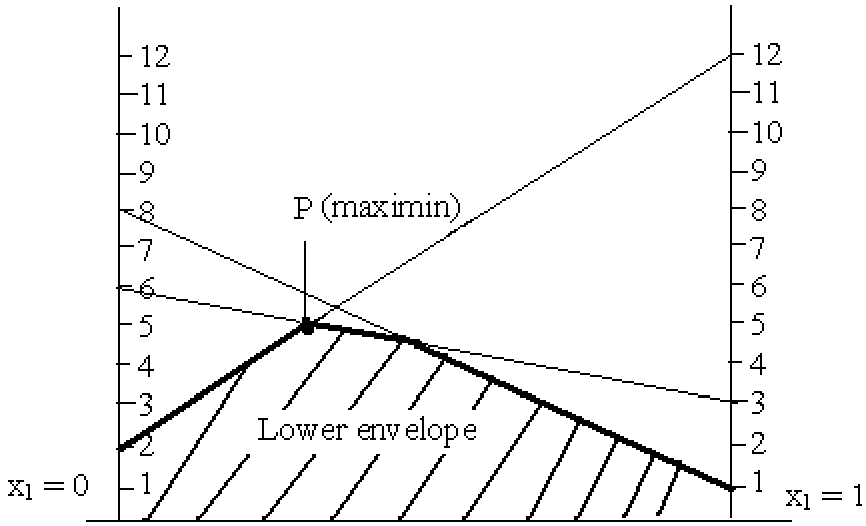
- Draw two vertical axes 1 unit apart. The two lines are $x_1 = 0$, $x_1 = 1$
- Take the points of the first row in the payoff matrix on the vertical line $x_1 = 1$ and the points of the second row in the payoff matrix on the vertical line $x_1 = 0$.
- The point a_{1j} on axis $x_1 = 1$ is then joined to the point a_{2j} on the axis $x_1 = 0$ to give a straight line. Draw 'n' straight lines for j=1, 2… n and determine the highest point of the lower envelope obtained. This will be the **maximin point**.
- The two or more lines passing through the maximin point determines the required 2 x 2 payoff matrix. This in turn gives the optimum solution by making use of analytical method.

Example 1

Solve by graphical method

$$\begin{array}{cc} & \begin{array}{ccc} B1 & B2 & B3 \end{array} \\ \begin{array}{c} A1 \\ A2 \end{array} & \begin{bmatrix} 1 & 3 & 12 \\ 8 & 6 & 2 \end{bmatrix} \end{array}$$

Solution



$$\begin{array}{c|cc|c} & B2 & B3 & \\ \hline A1 & 3 & 12 & 4 \\ A2 & 6 & 2 & 9 \\ \hline & 10 & 3 & \end{array} \qquad V = \frac{a_{11}a_{22} - a_{21}a_{12}}{(a_{11}+a_{22}) - (a_{12}+a_{21})} = \frac{6-72}{5-18}$$

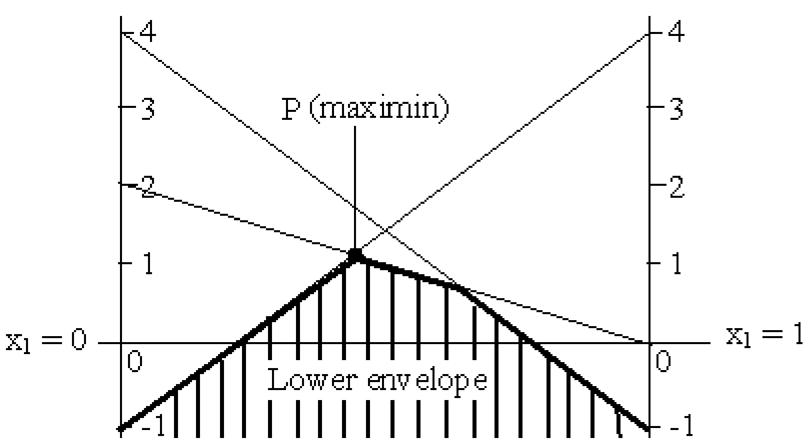
V = 66/13
S_A = (4/13, 9 /13)
S_B = (0, 10/13, 3 /13)

Example 2

Solve by graphical method

$$\begin{array}{c} \\ A1 \\ A2 \end{array} \begin{array}{c} \begin{array}{ccc} B1 & B2 & B3 \end{array} \\ \begin{bmatrix} 4 & -1 & 0 \\ -1 & 4 & 2 \end{bmatrix} \end{array}$$

Solution



$$\begin{array}{c|cc|c} & B1 & B3 & \\ \hline A1 & 4 & 0 & 3 \\ A2 & -1 & 2 & 4 \\ \hline & 2 & 5 & \end{array}$$

$$V = \frac{a_{11}\,a_{22} - a_{21}\,a_{12}}{(a_{11}+a_{22})-(a_{12}+a_{21})} = \frac{8-0}{6+1}$$

V = 8/7

S_A = (3/7, 4 /7)

S_B = (2/7, 0, 5 /7)

Algorithm for solving m x 2 matrix games

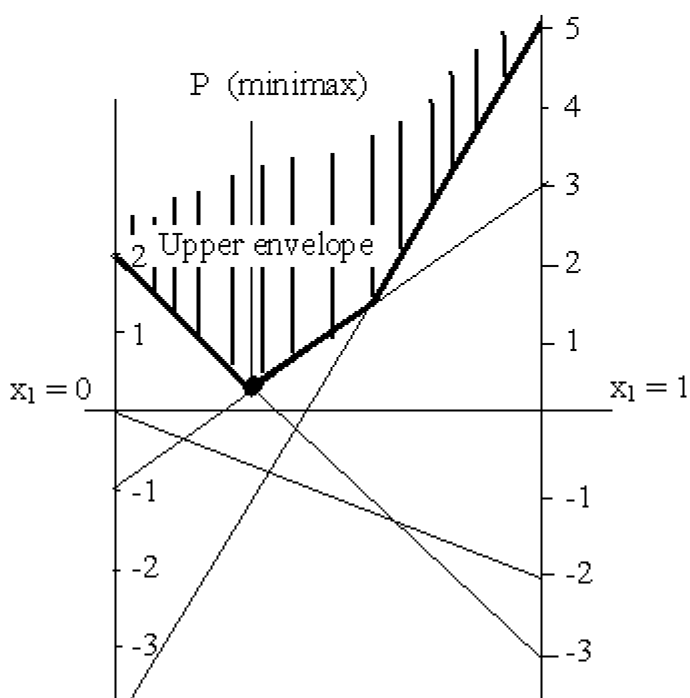
- Draw two vertical axes 1 unit apart. The two lines are x_1 =0, x_1 = 1 Take the points of the first row in the payoff matrix on the vertical line x_1 = 1 and the points of the second row in the payoff matrix on the vertical line x_1 = 0.
- The point a_{1j} on axis x_1 = 1 is then joined to the point a_{2j} on the axis x_1 = 0 to give a straight line. Draw 'n' straight lines for j=1, 2… n and determine the lowest point of the upper envelope obtained. This will be the **minimax point**.
- The two or more lines passing through the minimax point determines the required 2 x 2 payoff matrix. This in turn gives the optimum solution by making use of analytical method.

Example 1

Solve by graphical method

$$\begin{array}{c} \\ A1 \\ A2 \\ A3 \\ A4 \end{array}\begin{array}{c} \begin{array}{cc} B1 & B2 \end{array} \\ \begin{bmatrix} -2 & 0 \\ 3 & -1 \\ -3 & 2 \\ 5 & -4 \end{bmatrix} \end{array}$$

Solution



D

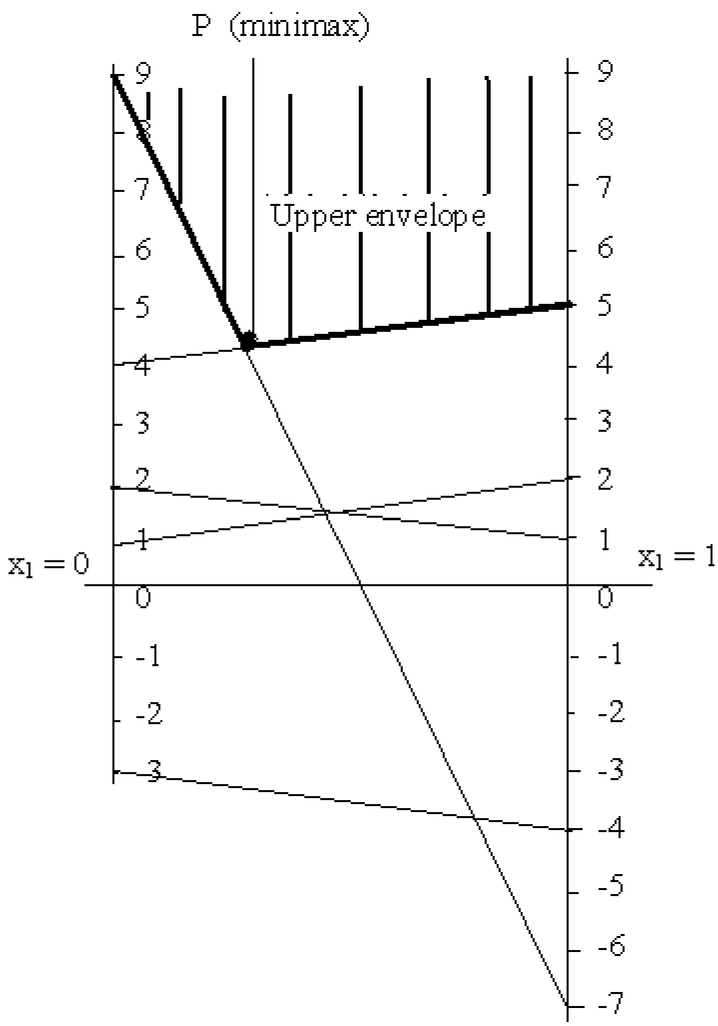
$V = 3/9 = 1/3$
$S_A = (0, 5/9, 4/9, 0)$
$S_B = (3/9, 6/9)$

Example 2

Solve by graphical method

$$\begin{array}{c} \\ A1 \\ A2 \\ A3 \\ A4 \\ A5 \end{array}\begin{array}{c} \begin{array}{cc} B1 & B2 \end{array} \\ \begin{bmatrix} 1 & 2 \\ 5 & 4 \\ -7 & 9 \\ -4 & -3 \\ 2 & 1 \end{bmatrix} \end{array}$$

Solution



$$\begin{array}{c|cc|c} & B1 & B2 & \\ \hline A2 & 5 & 4 & 16 \\ A3 & -7 & 9 & 1 \\ \hline & 5 & 12 & \end{array}$$

$$V = \frac{a_{11}a_{22} - a_{21}a_{12}}{(a_{11}+a_{22}) - (a_{12}+a_{21})} = \frac{45+28}{14+3}$$

V = 73/17
S_A = (0, 16/17, 1/17, 0, 0)
S_B = (5/17, 12 /17)

3.1.3 Simplex Method

Let us consider the 3 x 3 matrix

$$\begin{array}{c|ccc|} & B1 & B2 & B3 \\ \hline A1 & a_{11} & a_{12} & a_{13} \\ A2 & a_{21} & a_{22} & a_{23} \\ A3 & a_{31} & a_{32} & a_{33} \end{array}$$

As per the assumptions, A always attempts to choose the set of strategies with the non-zero probabilities say p_1, p_2, p_3 where $p_1 + p_2 + p_3 = 1$ that maximizes his minimum expected gain.

Similarly B would choose the set of strategies with the non-zero probabilities say q_1, q_2, q_3 where $q_1 + q_2 + q_3 = 1$ that minimizes his maximum expected loss.

Step 1
Find the minimax and maximin value from the given matrix

Step 2
The objective of A is to maximize the value, which is equivalent to minimizing the value 1/V. The LPP is written as

Min $1/V = p_1/V + p_2/V + p_3/V$
and constraints ≥ 1

It is written as

Min $1/V = x_1 + x_2 + x_3$
and constraints ≥ 1

Similarly for B, we get the LPP as the dual of the above LPP

Max $1/V = Y_1 + Y_2 + Y_3$
and constraints ≤ 1
Where $Y_1 = q_1/V$, $Y_2 = q_2/V$, $Y_3 = q_3/V$

Step 3
Solve the LPP by using simplex table and obtain the best strategy for the players

Example 1
Solve by Simplex method

$$\begin{array}{cc} & B \\ A & \begin{bmatrix} 3 & -2 & 4 \\ -1 & 4 & 2 \\ 2 & 2 & 6 \end{bmatrix} \end{array}$$

Solution

$$\begin{array}{ccc} & B & \\ A & \begin{bmatrix} 3 & -2 & 4 \\ -1 & 4 & 2 \\ 2 & 2 & 6 \end{bmatrix} & \begin{array}{l} -2 \\ -1 \\ 2 \text{ Maximin} \end{array} \\ & \begin{array}{ccc} 3 & 4 & 6 \\ \text{Minimax} & & \end{array} & \end{array}$$

We can infer that $2 \leq V \leq 3$. Hence it can be concluded that the value of the game lies between 2 and 3 and the $V > 0$.

LPP
Max $1/V = Y_1 + Y_2 + Y_3$
Subject to

$$3Y_1 - 2Y_2 + 4Y_3 \leq 1$$
$$-1Y_1 + 4Y_2 + 2Y_3 \leq 1$$
$$2Y_1 + 2Y_2 + 6Y_3 \leq 1$$
$$Y_1, Y_2, Y_3 \geq 0$$

SLPP
Max $1/V = Y_1 + Y_2 + Y_3 + 0s_1 + 0s_2 + 0s_3$
Subject to

$$3Y_1 - 2Y_2 + 4Y_3 + s_1 = 1$$
$$-1Y_1 + 4Y_2 + 2Y_3 + s_2 = 1$$
$$2Y_1 + 2Y_2 + 6Y_3 + s_3 = 1$$
$$Y_1, Y_2, Y_3, s_1, s_2, s_3 \geq 0$$

Basic Variables	C_B	Y_B	Y_1	Y_2	Y_3	S_1	S_2	S_3	Min Ratio Y_B / Y_K
	$C_j \rightarrow$		1	1	1	0	0	0	
S_1	0	1	3	-2	4	1	0	0	1/3→
S_2	0	1	-1	4	2	0	1	0	-
S_3	0	1	2	2	6	0	0	1	1/2
	1/V = 0		-1 ↑	-1	-1	0	0	0	
Y_1	1	1/3	1	-2/3	4/3	1/3	0	0	-
S_2	0	4/3	0	10/3	10/3	1/3	1	0	2/5
S_3	0	1/3	0	10/3	10/3	-2/3	0	1	1/10→
	1/V =1/3		0	-5/3 ↑	1/3	1/3	0	0	
Y_1	1	2/5	1	0	2	1/5	0	1/5	
S_2	0	1	0	0	0	1	1	-1	
Y_2	1	1/10	0	1	1	-1/5	0	3/10	
	1/V = 1/2		0	0	2	0	0	1/2	

1/V =1/2
V = 2

$y_1 = 2/5 * 2 = 4/5$
$y_2 = 1/10 * 2 = 1/5$
$y_3 = 0 * 2 = 0$

$x_1 = 0*2 = 0$
$x_2 = 0*2 = 0$
$x_3 = 1/2*2 = 1$

$S_A = (0, 0, 1)$
$S_B = (4/5, 1/5, 0)$
Value = 2

Example 2

$$\begin{array}{cc} & B \\ A & \begin{bmatrix} 1 & -1 & -1 \\ -1 & -1 & 3 \\ -1 & 2 & -1 \end{bmatrix} \end{array}$$

Solution

$$\begin{array}{cc} & B \\ A & \left[\begin{array}{ccc} 1 & -1 & -1 \\ -1 & -1 & 3 \\ -1 & 2 & -1 \end{array}\right] \begin{array}{c} -1 \\ -1 \\ -1 \end{array} \\ & \begin{array}{ccc} 1 & 2 & 3 \end{array} \end{array}$$

Maximin = -1
Minimax = 1

We can infer that $-1 \leq V \leq 1$

Since maximin value is -1, it is possible that value of the game may be negative or zero, thus the constant 'C' is added to all the elements of matrix which is at least equal to the negative of maximin.

Let C = 1, add this value to all the elements of the matrix. The resultant matrix is

$$\begin{array}{cc} & B \\ A & \left[\begin{array}{ccc} 2 & 0 & 0 \\ 0 & 0 & 4 \\ 0 & 3 & 0 \end{array}\right] \end{array}$$

LPP
Max $1/V = Y_1 + Y_2 + Y_3$
Subject to

$2Y_1 + 0Y_2 + 0Y_3 \leq 1$
$0Y_1 + 0Y_2 + 4Y_3 \leq 1$
$0Y_1 + 3Y_2 + 0Y_3 \leq 1$
$Y_1, Y_2, Y_3 \geq 0$

SLPP
Max $1/V = Y_1 + Y_2 + Y_3 + 0s_1 + 0s_2 + 0s_3$
Subject to

$2Y_1 + 0Y_2 + 0Y_3 + s_1 = 1$
$0Y_1 + 0Y_2 + 4Y_3 + s_2 = 1$
$0Y_1 + 3Y_2 + 0Y_3 + s_3 = 1$
$Y_1, Y_2, Y_3, s_1, s_2, s_3 \geq 0$

Basic Variables	C_B	Y_B	Y_1	Y_2	Y_3	S_1	S_2	S_3	Min Ratio Y_B / Y_K
	$C_j \rightarrow$		1	1	1	0	0	0	
S_1	0	1	2	0	0	1	0	0	1/2→
S_2	0	1	0	0	4	0	1	0	-
S_3	0	1	0	3	0	0	0	1	-
			↑						
	1/V =0		-1	-1	-1	0	0	0	
Y_1	1	1/2	1	0	0	1/2	0	0	-
S_2	0	1	0	0	4	0	1	0	-
S_3	0	1	0	3	0	0	0	1	1/3→
				↑					
	1/V =1/2		0	-1	-1	1/2	0	0	
Y_1	1	1/2	1	0	0	1/2	0	0	-
S_2	0	1	0	0	4	0	1	0	1/4→
Y_2	1	1/3	0	1	0	0	0	1/3	-
					↑				
	1/V = 5/6		0	0	-1	1/2	0	1/3	

Y_1	1	1/2	1	0	0	1/2	0	0
Y_3	1	1/4	0	0	1	0	1/4	0
Y_2	1	1/3	0	1	0	0	0	1/3
	1/V =13/12		0	0	0	1/2	1/4	1/3

1/V =13/12

V = 12/13

y_1 = 1/2 * 12/13 = 6/13 y_2 = 1/3 * 12/13 = 4/13 y_3 = 1/4 * 12/13 = 3/13

x_1 = 1/2*12/13 = 6/13 x_2 = 1/4 * 12/13 = 3/13 x_3 = 1/3 * 12/13 = 4/13

S_A = (6/13, 3/13, 4/13) S_B = (6/13, 4/13, 3/13)

Value = 12/13 – C =12/13 -1 = -1/13

INTRODUCTION

In the previous chapters we have dealt with problems where two or more competing candidates are in race for using the same resources and how to decide which candidate (product) is to be selected so as to maximize the returns (or minimize the cost).

Now let us look to a problem, where we have to determine the order or sequence in which the jobs are to be processed through machines so as to minimize the total processing time. Here the total effectiveness, which may be the time or cost that is to be minimized is the function of the order of sequence. Such type of problem is known as **SEQUENCING PROBLEM.**

In case there are three or four jobs are to be processed on two machines, it may be done by trial and error method to decide the optimal sequence (*i.e.* by method of enumeration). In the method of enumeration for each sequence, we calculate the total time or cost and search for that sequence, which consumes the minimum time and select that sequence. This is possible when we have small number of jobs and machines. But if the number of jobs and machines increases, then the problem becomes complicated. It cannot be done by method of enumeration. Consider a problem, where we have 'n' machines and 'm' jobs then we have $(n!)^m$ theoretically possible sequences. For example, we take $n = 5$ and $m = 5$, then we have $(5!)^5$ sequences *i.e.* which works out to 25, 000,000,000 possible sequences. It is time consuming to find all the sequences and select optima among all the sequences. Hence we have to go for easier method of finding the optimal sequence. Let us discuss the method that is used to find the optimal sequence. Before we go for the method of solution, we shall define the sequencing problem and types of sequencing problem. The student has to remember that the sequencing problem is basically a **minimization problem or minimization model.**

THE PROBLEM:(DEFINITION)

A general sequencing problem may be defined as follows:

Let there be 'n' jobs (J_1, J_2, J_3 ………J_n) which are to be processed on 'm' machines (A, B, C, ………), where the order of processing on machines *i.e.* for example, *ABC* means first on machine A, second on machine *B* and third on machine *C* or *CBA* means first on machine *C*, second on machine *B* and third on machine *A* etc. and the processing time of jobs on machines (actual or expected) is known to us, then our job is to find the optimal sequence of processing jobs that minimizes the total processing time or cost. Hence our job is to find that sequence out of $(n!)^m$ sequences, which minimizes the total elapsed time (*i.e.*. time taken to process all the jobs). The usual notations used in this problem are:

A_i = Time taken by *i* th job on machine *A* where *i* = I, 2,3…*n*. Similarly we can interpret for machine *B* and *C* *i.e.* B_i and C_i etc.

T = Total elapsed time which includes the idle time of machines if any and set up time and transfer time.

Assumptions Made in Sequencing Problems

Principal assumptions made for convenience in solving the sequencing problems are as follows:

(a) The processing times A_i and B_i etc. are exactly known to us and they are independent of order of processing the job on the machine. That is whether job is done first on the machine, last on the machine, the time taken to process the job will not vary it remains constant.

(b) The time taken by the job from one machine to other after processing on the previous machine is negligible. (Or we assume that the processing time given also includes the transfer time and setup time).

(c) Each job once started on the machine, we should not stop the processing in the middle. It is to be processed completely before loading the next job.

(d) The job starts on the machine as soon as the job and the machine both become idle (vacant). This is written as **job is next to the machine and the machine is next to the job.** (This is exactly the meaning of transfer time is negligible).

(e) No machine may process more than one job simultaneously. (This means to say that the job once started on a machine, it should be done until completion of the processing on that machine).

(f) The cost of keeping the semi-finished job in inventory when next machine on which the job is to be processed is busy is assumed to be same for all jobs or it is assumed that it is too small and is negligible. That is in process inventory cost is negligible.

(g) While processing, no job is given priority *i.e.* the order of completion of jobs has no significance. The processing times are independent of sequence of jobs.

(h) There is only one machine of each type.

Applicability

The sequencing problem is very much common in Job workshops and Batch production shops. There will be number of jobs which are to be processed on a series of machine in a specified order depending on the physical changes required on the job. We can find the same situation in computer center where number of problems waiting for a solution. We can also see the same situation when number of critical patients waiting for treatment in a clinic and in Xerox centers, where number of jobs is in queue, which are to be processed on the Xerox machines. Like this we may find number of situations in real world.

Types of Sequencing Problems

There are various types of sequencing problems arise in real world. All sequencing problems cannot be solved. Though mathematicians and Operations Research scholars are working hard on the problem satisfactory method of solving problem is available for few cases only. The problems, which can be solved, are:

(i) 'n' jobs are to be processed on two machines say machine A and machine B in the order AB. This means that the job is to be processed first on machine A and then on machine B.

(j) 'n' jobs are to be processed on three machines A,B and C in the order ABC i.e. first on machine A, second on machine B and third on machine C.

(k) 'n' jobs are to be processed on 'm' machines in the given order

(l) Two jobs are to be processed on 'm' machines in the given order.

SOLUTIONS FOR SEQUENCING PROBLEMS

Now let us take above mentioned types problems and discuss the solution methods.

'N' Jobs and Two Machines

If the problem given has two machines and two or three jobs, then it can be solved by using the Gantt chart. But if the numbers of jobs are more, then this method becomes less practical. (For understanding about the Gantt chart, the students are advised to refer to a book on Production and Operations Management (chapter on Scheduling).

Gantt chart consists of *X* -axis on which the time is noted and *Y*-axis on which jobs or machines are shown. For each machine a horizontal bar is drawn. On these bars the processing of jobs in given sequence is marked. Let us take a small example and see how Gantt chart can be used to solve the same.

1. Problem

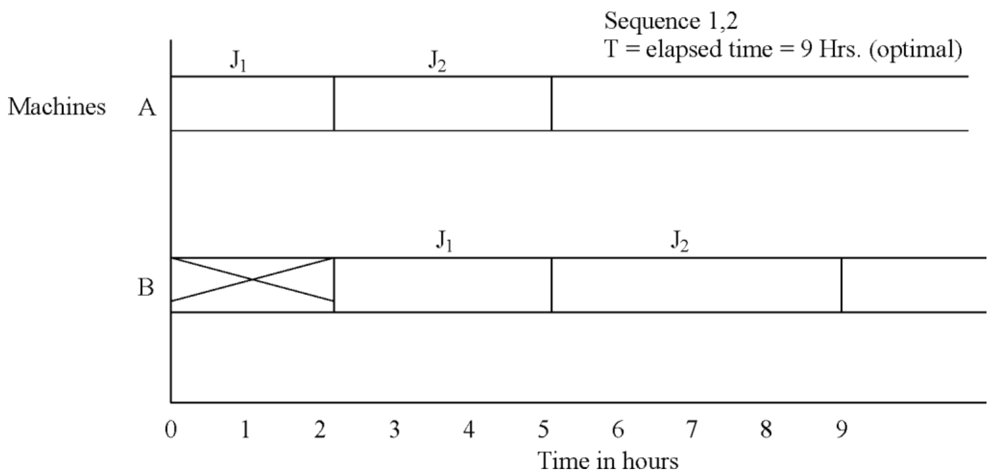
There are two jobs job 1 and job 2. They are to be processed on two machines, machine *A* and Machine *B* in the order *AB*. Job 1 takes 2 hours on machine *A* and 3 hours on machine *B*. Job 2 takes 3 hours on machine *A* and 4 hours on machine *B*. Find the optimal sequence which minimizes the total elapsed time by using Gantt chart.

2. Solution

Jobs.	*Machines (Time in hours)*	
	A	B
1	2	3
2	3	4

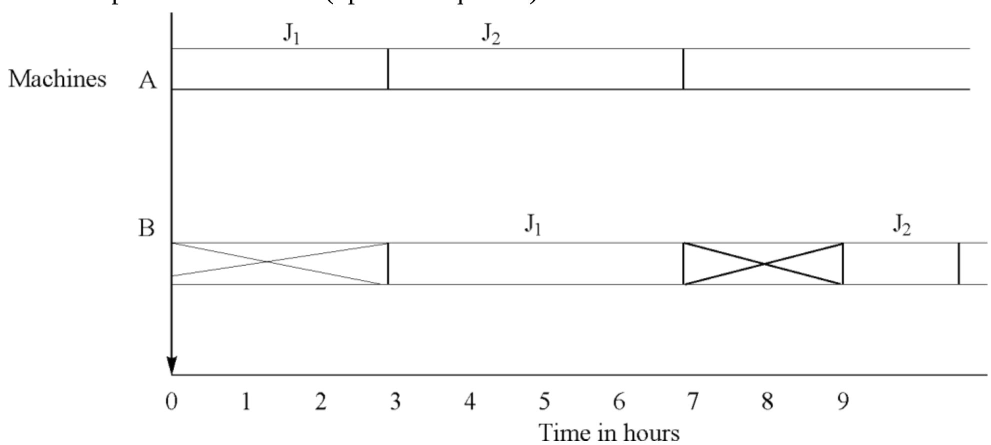
(m) Total elapsed time for sequence 1,2 *i.e.* first job 1 is processed on machine *A* and then on second machine and so on.

Draw *X* - axis and Y- axis, represent the time on *X* - axis and two machines by two bars on *Y*- axis. Then mark the times on the bars to show processing of each job on that machine.



Sequence 1,2

Total = elapsed time = 9 Hrs. (optimal sequence)



Gantt chart.

Both the sequences shows the elapsed time = 9 hours.

The draw back of this method is for all the sequences, we have to write the Gantt chart and find the total elapsed times and then identify the optimal solution. This is laborious and time consuming. If we have more jobs and more machines, then it is tedious work to draw the chart for all sequences. Hence we have to go for analytical methods to find the optimal solution without drawing charts.

Analytical Method

A method has been developed by **Johnson and Bellman** for simple problems to determine a sequence of jobs, which minimizes the total elapsed time. The method:

1. 'n' jobs are to be processed on two machines A and B in the order AB (*i.e*. each job is to be processed first on A and then on B) and passing is not allowed. That is which ever job is processed first on machine A is to be first processed on machine B also, Which ever job is processed second on machine A is to be processed second on machine B also and so on. That means each job will first go to machine A get processed and then go to machine B and get processed. ***This rule is known as no passing rule.***
2. Johnson and Bellman method concentrates on minimizing the idle time of machines. Johnson and Bellman have proved that optimal sequence of 'n' jobs which are to be processed on two machines A and B in the order AB necessarily involves the same ordering of jobs on each machine. This result also holds for three machines but does not necessarily hold for more than three machines. Thus total elapsed time is minimum when the sequence of jobs is same for both the machines.
3. Let the number of jobs be 1,2,3,… ....n

The processing time of jobs on machine A be A_1, A_2, A_3 .... A_n
The processing time of jobs on machine B be B_1, B_2, B_3 .... B_n

Jobs	*Machining time in hours.*		
	Machine A	*Machine B*	*(Order of processing is AB)*
1	A_1	B_1	
2	A_2	B_2	
3	A_3	B_3	
………	……………………	………………..	
I	A_I	B_I	
………	……………………	………………….	
S	A_S	B_S	
………	……………………	………………..	
………	……………………	………………..	
T	A_T	B_T	
………	……………………	……………….	
………	……………………	………………..	
N	A_N	B_N	

4. Johnson and Bellman algorithm for optimal sequence states *that* **identify the smallest element in the given matrix.** ***If the smallest element falls under column 1 i.e under machine I then do that job first***. As the job after processing on machine 1 goes to machine 2, it reduces the idle time or waiting time of machine 2. ***If the smallest element falls under column 2 i.e under machine 2 then do that job last***. This reduces the idle time of machine 1. *i.e*. if r th job is having smallest element in first column, then do the r^{th} job first. If s th job has the smallest element, which falls under second column, then do the s th job last. Hence the basis for Johnson and Bellman method is to keep the idle time of machines as low as possible. Continue the above process until all the jobs are over.

1	2	3		n–1	n
r					s

5. If there are 'n' jobs, first write 'n' number of rectangles as shown. When ever the smallest elements falls in column 1 then enter the job number in first rectangle. If it falls in second column, then write the job number in the last rectangle. Once the job number is entered, the second rectangle will become first rectangle and last but one rectangle will be the last rectangle.
6. Now calculate the total elapsed time as discussed. Write the table as shown. Let us assume that the first job starts at Zero th time. Then add the processing time of job (first in the optimal sequence) and write in out column under machine 1. This is the time when the first job in the optimal sequence leaves machine 1 and enters the machine 2. Now add processing time of job on machine 2. This is the time by which the processing of the job on two machines over. Next consider the job, which is in second place in optimal sequence. This job enters the machine 1 as soon the machine becomes vacant, i.e first job leaves to second machine. Hence enter the time in out column for first job under machine 1 as the starting time of job two on machine 1. Continue until all the jobs are over. Be careful to see that whether the machines are vacant before loading. Total elapsed time may be worked out by drawing Gantt chart for the optimal sequence.
7. Points to remember:

(a) If there is tie i.e we have smallest element of same value in both columns, then:

(i) Minimum of all the processing times is A_r which is equal to B_s *i.e.* Min (A_i, B_i) = A_r = B_s then do the r th job first and s th job last.

(ii) If Min (A_i, B_i) = A_r and also $A_r = A_k$ (say). Here tie occurs between the two jobs having same minimum element in the same column *i.e.* first column we can do either r th job or k th job first. There will be two solutions. When the ties occur due to element in the same column, then the problem will have alternate solution. If more number of jobs have the same minimum element in the same column, then the problem will have many alternative solutions. If we start writing all the solutions, it is a tedious job. Hence it is enough that the students can mention that the problem has alternate solutions. The same is true with B_i s also. If more number of jobs have same minimum element in second column, the problem will have alternate solutions.

3. Problem

There are five jobs, which are to be processed on two machines *A* and *B* in the order *AB*. The processing times in hours for the jobs are given below. Find the optimal sequence and total elapsed time. ***(Students has to remember in sequencing problems if optimal sequence is asked, it is the duty of*** the student to find the total elapsed time also).

Jobs:	1	2	3	4	5
Machine A (Time in hrs.)	2	6	4	8	10
Machine B (Time in Hrs)	3	1	5	9	7

The smallest element is 1 it falls under machine *B* hence do this job last i.e in 5 th position. Cancel job 2 from the matrix. The next smallest element is 2, it falls under machine A hence do this job first, i.e in the first position. Cancel the job two from matrix. Then the next smallest element is 3 and it falls under machine *B*. Hence do this job in fourth position. Cancel the job one from the matrix. Proceed like this until all jobs are over.

1	3	4	5	2

1. Total elapsed time:

OPTIMAL	*MACHINE -A*		*MACHINE - B*		*MACHINE IDLE JOB IDLE*			*REMARKS*
SEQUENCE	*IN*	*OUT*	*IN*	*OUT*	*A*	*B*		
1	0	2	2	5		2		
3	2	6	6	11		1		As the Machine B Finishes Work at 5 Th hour will be Idle for 1 Hour.
4	6	14	14	23		3		-do- 3 hr.
5	14	24	24	31		1		-do- 1 hr.
2	24	30	31	32	1	2		1 hr as job finished early 1 hr idle.

Total elapsed time = 32 hours. (This includes idle time of job and idle time of machines).

The procedure: Let Job 1 is loaded on machine *A* first at zero th time. It takes two hours to process on the machine. Job 1 leaves the machine *A* at two hours and enters the machine 2 at 2-nd hour. Up to the time i.e first two hours, the machine *B* is idle. Then the job 1 is processed on machine *B* for 3 hours and it will be unloaded. As soon as the machine *A* becomes idle, *i.e*. at 2 nd hour then next job 3 is loaded on machine *A*. It takes 4 hours and the job leaves the machine at 6 th hour and enters the machine *B* and is processed for 6 hours and the job is completed by 11 th hour. (Remember if the job is completed early and the Machine *B* is still busy, then the job has to wait and the time is entered in job idle column. In case the machine B completes the previous job earlier, and the machine *A* is still processing the next job, the machine has to wait for the job. This will be shown as machine idle time for machine *B*.). Job 4 enters the machine *A* at 6 th hour and processed for 8 hours and leaves the machine at 14 th hour. As the machine *B* has finished the job 3 by 11 th hour, the machine has to wait for the next job (job 4) up to 14 th hour. Hence 3 hours is the idle time for the machine *B*. In this manner we have to calculate the total elapsed time until all the jobs are over.

4. Problem

There are 6 jobs to be processed on Machine *A*. The time required by each job on machine *A* is given in hours. Find the optimal sequence and the total time elapsed.

Job:	1	2	3	4	5	6
Time in hours. Machine A	6	4	3	2	9	8

5. Solution

Here there is only one machine. Hence the jobs can be processed on the machine in any sequence depending on the convenience. The total time elapsed will be total of the times given in the problem. As soon as one job is over the other follows. The total time is 32 hours. The sequence may be any order. For example: 1,2,3,4,5,6 or 6,5,4,3,2,1, or 2, 4 6 1 3 5 and so on.

6. Problem

A machine operator has to perform two operations, turning and threading, on a number of different jobs. The time required to perform these operations in minutes for each job is given. Determine the order in which the jobs should be processed in order to minimize the total time required to turn out all the jobs.

Jobs:	1	2	3	4	5	6
Time for turning (in min.)	3	12	5	2	9	11
Time for threading (in min).	8	10	9	6	3	1

7. Solution

The smallest element is 1 in the given matrix and falls under second operation. Hence do the 6 th job last. Next smallest element is 2 for the job 4 and falls under first operation hence do the fourth job first. Next smallest element is 3 for job 1 falls under first operation hence do the first job second. Like this go on proceed until all jobs are over. The optimal sequence is :

4	1	3	2	5	6

Optimal sequence.	*Turning operation*		*Threading operation*		*Job idle*	*Machine idle.*	
	In	*out*	*In*	*out*		*Turning*	*threading.*
4	0	2	2	8	------		2
1	2	5	8	16	3		
3	5	10	16	25	6		
2	10	22	25	35	3		
5	22	31	35	38	4		
6	31	42	42	43	--	1	----
	Total elapsed time:		**43minutes.**				

The Job idle time indicates that there must be enough space to store the in process inventory between two machines. This point is very important while planning the layout of machine shops.

8. Problem

There are seven jobs, each of which has to be processed on machine *A* and then on Machine *B* (order of machining is *AB*). Processing time is given in hours. Find the optimal sequence in which the jobs are to be processed so as to minimize the total time elapsed.

JOB:	1	2	3	4	5	6	7
MACHINE: A (TIME IN HOURS).	3	12	15	6	10	11	9
MACHINE: B (TIME IN HOURS).	8	10	10	6	12	1	3

9. Solution

By Johnson and Bellman method the optimal sequence is:

1	4	5	3	2	7	6.

Optimal Sequence	Machine:A		Machine:B		Machine idle time		Job idle time	Remarks.
Squence	In	out	In	Out	A	B		
1	0	3	3	11		3	–	
4	3	9	11	17			2	Job finished early
5	9	19	19	31		2		Machine A take more time.
3	19	34	34	44		3		Machine A takes more time.
2	34	46	46	56		2		- do-
7	46	55	56	59			1	Job finished early.
6	55	66	66	67	1	7		Machine A takes more time. Last is finished on machine A at 66 th hour.
	Total		**Elapsed**		**Time = 67 hours.**			

Problem

Find the optimal sequence that minimizes the total elapsed time required to complete the following tasks on two machines I and II in the order first on Machine I and then on Machine II.

Task:	A	B	C	D	E	F	G	H	I
Machine I (time in hours).	2	5	4	9	6	8	7	5	4
Machine II (time in hours).	6	8	7	4	3	9	3	8	11

Solution

By Johnson and Bellman method we get two sequences (this is because both machine B and H are having same processing times).

The two sequences are:

A	C	I	B	H	F	D	G	E.
A	C	I	H	B	F	D	G	E

Sequence	*Machine I*		*Machine II*		*Machine Idle*		*Job idle*	*Remarks.*
	In	*out*	*In*	*Out*	*I*	*II*		
A	O	2	2	8		2		
C	2	6	8	15			2	Job on machine I finished early.
I	6	10	15	26			5	Do
B	10	15	26	34			11	Do
H	15	20	34	42			14	Do
F	20	28	42	51			14	Do
D	28	37	51	55			14	Do
G	37	44	55	58			11	Do
E	44	50	58	61	11		8	Do.And machine I finishes its work at 50th hour.
	Total		**Elapsed time : 61 hours.**					

Problem 6.7.

A manufacturing company processes 6 different jobs on two machines A and B in the order AB. Number of units of each job and its processing times in minutes on A and B are given below. Find the optimal sequence and total elapsed time and idle time for each machine.

Job Number	*Number of units of each job.*	*Machine A: time in minutes.*	*Machine B: time in minutes.*
1	3	5	8
2	4	16	7
3	2	6	11
4	5	3	5
5	2	9	7.5
6	3	6	14

Solution

The optimal sequence by using Johnson and Bellman algorithm is

Sequence:	4	1	3	6	5	2
Number of units.	5	3	2	3	2	4

First do the 5 units of job 4, Second do the 3 units of job 1, third do the 2 units of job 3, fourth process 3 units of job 6, fifth process 2 units of job 5 and finally process 4 units of job 2.

Sequence of jobs	*Number. of units of job*	*Machine A Time in mins*		*Machine B Time in mins.*		*Idle time of machines*		*Job idle.*	*Remarks.*
		In	*out*	*In*	*out*	*A*	*B*		
4	1 st.	0	3	3	8	--	3	–	–
	2 nd	3	6	8	13				
	3 rd.	6	9	13	18				
	4 th	9	12	18	23				
	5th	12	15	23	28				
1	1 st	15	20	28	36			8	Machine B
	2 nd	20	25	36	44				BecomesVacant
	3rd	25	30	44	52				at 8th min.
3	1 st	30	36	52	63			16	Do (52 nd min.)
	2 nd.	36	42	63	74				
6	1 st.	42	48	74	88			26	Do (74 th min.)
	2 nd	48	54	88	102				
	3 rd	54	60	102	116				
5	1 st	60	69	116	123.5			47	Do (116 th min.)
	2 nd.	69	78	123.5	131				
2	1 st	78	94	131	138			37	Do (131 th min.)
	2 nd.	94	110	138	145				
	3 rd	110	126	145	152				
	4 th	126	142	152	159	17			
		Total Elapsed		**Time = 159 min**					

Total elapsed time = 159 mins. Idle time for Machine A = 17 mins. And that for machine B is 3 mins

SEQUENCING OF 'N' JOBS ON THREE MACHINES

When there are '*n*' jobs, which are to be processed on three machines say *A*, *B*, and *C* in the order *ABC i.e* first on machine *A*, second on machine *B* and finally on machine *C*. We know processing times in time units. As such there is no direct method of sequencing of '*n*' jobs on three machines. Before solving, **a three-machine problem is to be converted into a two-machine problem**. The procedure for converting a three-machine problem into two-machine problem is:

(a) Identify the smallest time element in the first column, *i.e.* for machine 1 let it be A_r.
(b) Identify the smallest time element in the third column, *i.e.* for machine 3, let it be C_s
(c) Identify the highest time element in the second column, *i.e.* for the center machine, say machine 2, let it be B_i.
(d) Now minimum time on machine 1 *i.e.* A_r must be ≥ maximum time element on machine 2, *i.e.* B_i

OR

Minimum time on third machine *i.e.* C_s must be ≥ maximum time element on machine 2 *i.e.* B_i

OR

Both A_r and C_s must be ≥ B_i
(e) If the above condition satisfies, then we have to workout the time elements for two hypothetical machines, namely machine *G* and machine *H*. The time elements for machine *G*, $G_i = A_i + B_i$.
The time element for machine *H*, is $H_i = B_i + C_i$
(f) Now the three-machine problem is converted into two-machine problem. We can find sequence by applying Johnson Bellman rule.
(g) All the assumption mentioned earlier will hold good in this case also.

Problem

A machine operator has to perform three operations, namely plane turning, step turning and taper turning on a number of different jobs. The time required to perform these operations in minutes for each operating for each job is given in the matrix given below. Find the optimal sequence, which minimizes the time required.

Job.	*Time for plane turning In minutes*	*Time for step turning in minutes*	*Time for taper turning. in minutes.*
1	3	8	13
2	12	6	14
3	5	4	9
4	2	6	12
5	9	3	8
6	11	1	13

Solution

Here Minimum $A_i = 2$, Maximum $B_i = 8$ and Minimum $C_i = 8$.

As the maximum $B_i = 8 =$ Minimum C_i, we can solve the problem by converting into two-machine problem.

Now the problem is:

Job	*Machine G* $(A_i + B_i)$ *Minutes.*	*Machine H* $(B_i + H_i)$ *Minutes.*
1	11	21
2	18	20
3	9	13
4	8	18
5	12	11
6	12	14

By applying Johnson and Bellman method, the optimal sequence is:

4	3	1	6	5	2

Now we can work out the Total elapsed time as we worked in previous problems.

Sequence	Plane turning Time in min.		Step turning Time in min.		Taper turning Time in Min.		Job Idle Time in Min.	Machine idle Time in Min.	Remarks.
	In	out	In	out	In	out		Tu StTu Tap Tu	
4	0	2	2	8	8	20		2 8	Until first Job comes 2nd and 3rd Operations idle.
3	2	7	8	12	20	29	1 + 8		
1	7	10	12	20	29	42	2 + 9		
6	10	21	21	22	42	55	20	1	
2	21	33	33	39	55	69	16	11	
5	33	42	42	45	69	77	14	3	
	Total		**Elapsed**		**Time: 77 min.**				

10. *Problem 6.9.*

There are 5 jobs each of which is to be processed on three machines *A*, *B*, and *C* in the order *ACB*. The time required to process in hours is given in the matrix below. Find the optimal sequence.

Job:	1	2	3	4	5
Machine A:	3	8	7	5	4
Machine B:	7	9	5	6	10
Machine C:	4	5	1	2	3.

Solution

Here the given order is *ACB*. *i.e.* first on machine *A*, second on Machine *C* and third on Machine *B*. Hence we have to rearrange the machines. Machine *C* will become second machine. Moreover optimal sequence is asked. But after finding the optimal sequence, we have to work out total elapsed time also. The procedure is first rearrange the machines and convert the problem into two-machine problem if it satisfies the required condition. Once it is converted, we can find the optimal sequence by applying Johnson and Bellman rule.

The problem is:

Job:	1	2	3	4	5
Machine A:	3	8	7	5	4
Machine C:	4	5	1	2	3
Machine B:	7	9	5	6	10

Max A_i = 8 Hrs. , Max B_i (third machine) = 5 Hrs. and minimum C_i = Middle machine = 5 Hrs. As Max B_i = Min C_i = 5, we can convert the problem into 2- machine problem.

Two-machine problem is:

Job:	1	2	3	4	5
Machine *G*: (*A* + *C*)	7	13	8	7	7
Machine *H*: (*C* + *B*)	11	14	6	8	13

By applying, Johnson and Bellman Rule, the optimal sequence is: We find that there are alternate solutions, as the elements 7 and 8 are appearing more than one time in the problem.

The solutions are:

4	1	5	2	3

4	5	1	2	3

1	4	5	2	3

5	1	4	2	3

5	4	1	2	3

Let us work out the total time elapsed for any one of the above sequences. Students may try for all the sequence and they find that the total elapsed time will be same for all sequences.

Sequence.	*Machine A Time in Hrs.*		*Machine C Time in Hrs.*		*Machine B Time in Hrs.*		*Job idle. Time in Hrs*	*Machine Idle. Time in Hrs.*		
	In	*out*	*In*	*out*	*In*	*out*		*A*	*C*	*B*
4	0	5	5	7	7	13			5	7
1	5	8	8	12	13	20	1		1	
5	8	12	12	15	20	30	5			
2	12	20	20	25	30	39	5		5	
3	20	27	27	28	39	44	11	17	2+16	
	Total		**Elapsed**		**Time:44 Hrs.**					

Total elapsed time = 44 hours. Idle time for Machine *A* is 17 hours. For machine *C* = 29 hrs and that for machine *B* is 7 hours.

Problem .

A ready-made dress company is manufacturing its 7 products through two stages *i.e.* cutting and Sewing. The time taken by the products in the cutting and sewing process in hours is given below:

Products:	1	2	3	4	5	6	7
Cutting:	5	7	3	4	6	7	12
Sewing:	2	6	7	5	9	5	8

(a) Find the optimal sequence that minimizes the total elapsed time.

(b) Suppose a third stage of production is added, namely Pressing and Packing, with processing time for these items as given below:

Product:	1	2	3	4	5	6	7
Pressing and Packing: (Time in hrs)	10	12	11	13	12	10	11

Find the optimal sequence that minimizes the total elapsed time considering all the three stages.

Solution

(*a*) Let us workout optimal sequence and total elapsed time for first two stages:

By Johnson and Bellman rule, the optimal sequence is:

3	4	5	7	2	6	1

Total Elapsed time:

Sequence	*Cutting Dept. Time in Hrs.*		*Sewing dept. Time in Hrs.*		*Job idle Time in Hrs.*	*Machine idle. Time in Hrs.*		*Remarks.*
	In	*out*	*In*	*out*		*Cutting*	*Sewing.*	
3	0	3	3	10			3	Sewing starts after cutting.
4	3	7	10	15	3			
5	7	13	15	24	2			
7	13	25	25	33			1	
2	25	32	33	39	1			
6	32	39	39	44				
1	39	44	44	46		2		
	Total		**Elapsed**		**Time in Hrs. = 46 Hrs.**			

Total elapsed time is 46 Hrs. Idle time for cutting is 2 Hrs, and that for Sewing is 4 Hrs.

a) When the Pressing and Packing department is added to Cutting and Sewing, the problem becomes '*n*' jobs and 3-machine problem. We must check whether we can convert the problem into 2- machine problem.

The problem is

Products:	1	2	3	4	5	6	7
Cutting dept. (Hrs):	5	7	3	4	6	7	12
Sewing dept (Hrs);	2	6	7	5	9	5	8
Pressing and Packing dept. (Hrs.):	10	12	11	13	12	10	11

Minimum time element for first department is 3 Hrs. and that for third department is 10 Hrs. And maximum time element for second department i.e sewing department is 9 Hrs. As the minimum time element of third department is greater than that of minimum of second department, we can convert the problem into 2-machine problem.

Now 7 jobs and 2- machine problem is:

Product:	1	2	3	4	5	6	7
Department G (= Cutting + Sewing):	7	13	10	9	15	12	20
Department H (= Sewing + Packing):	12	18	18	18	21	15	19

By Johnson and Bellman rule the optimal sequence is:

1	4	3	6	2	5	7

Sequence	Cutting Dept. Time in Hrs.		Sewing Dept. Time in Hrs.		Packing dept. Time in Hrs.		Job idle. Time in Hrs.	Dept. Idle Time in Hrs..			Remarks.
	In	out	In	out	In	out		Cut	Sew	Pack.	
1	0	5	5	7	7	17			5	7	
4	5	9	9	14	17	30	3		2		
3	9	12	14	21	30	41	2 + 9				
6	12	19	21	26	41	51	2 + 15				
2	19	26	26	32	51	63	19				
5	26	32	32	41	63	75	22				
7	32	44	44	52	75	86		42	3+34		
	Total		**ElapsedTime**		**= 86 Hrs**.						

Total elapsed time = 86 Hrs. Idle time for Cutting dept. is 42 Hrs. Idle time for sewing dept, is 44 Hrs. and for packing dept. it is 7 hrs.

(Point to note: The Job idle time shows that enough place is to be provided for in process inventory and the machine or department idle time gives an indication to production planner that he can load the machine or department with any job work needs the service of the machine or department. Depending on the quantum of idle time he can schedule the job works to the machine or department).

Processing of 'N' Jobs on 'M' Machines: (Generalization of 'n' Jobs and 3 -machine problem)

Though we may not get accurate solution by generalizing the procedure of '*n*' jobs and 3- machine problem to '*n*' jobs and '*m*' machine problem, we may get a solution, which is nearer to the optimal solution. In many practical cases, it will work out. The procedure is :

A general sequencing problem of processing of '*n*' jobs through '*m*' machines M_1, M_2, M_3, ………M_{n-1} , M_n in the order M_1, M_2, M_3 …… M_{n-1}, M_n can be solved by applying the following rules.

If a_{ij} where I = 1,2,3…..*n* and j = 1, 2, 3……….*m* is the processing time of *i th* job on *j th* machine, then find Minimum a_{i1} and Min. a_{im} (*i.e.* minimum time element in the first machine and

I *i*

minimum time element in last

Machine) and find Maximum a_{ij} of intermediate machines *i.e* 2 nd machine to m-1 machine.

i

The problem can be solved by converting it into a two-machine problem if the following conditions are satisfied.

(n) $\underset{i}{\text{Min}}\ a_{i1} \geq \underset{i}{\text{Max.}}\ a_{ij}$ for all j = 1,2,3,…..m-1

(o) $\underset{i}{\text{Min}}\ a_{im} \geq \underset{i}{\text{Max}}\ a_{ij}$ for all j = 1, 2,3 …….m-1

At least one of the above must be satisfied. Or both may be satisfied. If satisfied, then the problem can be converted into 2- machine problem where Machine $G = a_{i1} + a_{i2} + a_{i3} + \ldots\ldots\ldots\ldots + a_{i\,m-1}$ and

Machine $G = a_{i2} + a_{i3} + \ldots\ldots\ldots + a_{im}$. Where i = 1,2,3,….n.

Once the problem is *a* 2- machine problem, then by applying Johnson Bellman algorithm we can find optimal sequence and then workout total elapsed time as usual.

1. (Point to remember: Suppose $a_{i2} + a_{i3} + \ldots\ a_{i\,m-1}$ = *a* constant number for all consider two extreme machines *i.e.* machine 1 and machine -m as two machines and workout optimal sequence).

Problem .

There are 4 jobs *A*, *B*, *C* and *D*, which is to be, processed on machines M_1, M_2, M_3 and M_4 in the order $M_1\ M_2\ M_3\ M_4$ .The processing time in hours is given below. Find the optimal sequence.

Job	*Machine (Processing time in hours)*			
	M_1	M_2	M_3	M_4
	a_{i1}	a_{i2}	a_{i3}	a_{i4}
A	15	5	4	14
B	12	2	10	12
C	13	3	6	15
D	16	0	3	19

Solution

From the data given, Min a_{i1} is 12 and Min a_{i4} is 12.

Max a_{i2} = 5 and Max a_{i3} = 10.

As Min a_{i1} is > than both Min a_{i2} and Min a_{i3}, the problem can be converted into 2 – machine problem as discussed above. Two-machine problem is:

Jobs.	*Machines (Time in hours)*	
	G	*H*
A	15+5+4 = 29	5+4+14 = 23
B	12+2+10 = 24	2+10+12 = 24
C	13+3+6 = 22	3+6+15 = 24
D	16+0+3 = 19	0+3+19 = 22

Applying Johnson and Bellman rule, the optimal sequence is:

D	*C*	*B*	*A*

Total elapsed time:

Sequence	*Machine M_1 Time in hours*		*Machine M_2 Time in hours*		*Machine M_3 Time in hours*		*Machine M_4 Time in hours.*		*Job idle Time in hours.*	*Machine idle Time in hours.*			
	In	*out*	*In*	*out*	*in*	*out*	*In*	*out*		M_1	M_2	M_3	M_4
D	0	16	16	16	16	19	19	38				16	19
C	19	29	29	32	32	38	38	53			29	13	
B	29	41	41	43	43	53	53	65			9	5	
A	41	56	56	61	61	65	65	79		23	18	14	
	Total		**Elapsed**		**Time= 79 hrs.**								

Total Elapsed time = 79 hours.

Problem .

In a maintenance shop mechanics has to reassemble the machine parts after yearly maintenance in the order *PQRST* on four machines *A*, *B*, *C* and *D*. The time required to assemble in hours is given in the matrix below. Find the optimal sequence.

Machine.	*Parts (Time in hours to assemble)*				
	P	Q	R	S	T
A	7	5	2	3	9
B	6	6	4	5	10
C	5	4	5	6	8
D	8	3	3	2	6

Solution

Minimum assembling time for component *P* = 5 hours. Minimum assembling time for component *T* = 6 hours. And Maximum assembling time for components *Q*, *R*, *S* are 6 hrs, 5 hrs and 6 hours respectively.

This satisfies the condition required for converting the problem into 2 - machine problem. The two-machine problem is:

Machine	*Component G*	*Component H*	*(Condition: Minimum P_i > Maximum Q_i, R_i,*
	(Time in hours)		*and Si. OR*
	(P+Q+R+S)	*(Q+R+S+T)*	*Minimum T_i > Maximum Q_i, R_i, and S_i.*
A	17	19	
B	21	25	
C	20	23	
D	16	14	

The optimal sequence by applying Johnson and Bellman rule is:

A	C	B	D

Total Elapsed Time:

Sequence	*Component P Time in hours*		*Component Q Time in hours*		*Component R Time in hours.*		*Component S Time in hours*		*Component T Time in hours.*		*Men idle Hrs P Q,R,S,T*	*Job idle Hrs.*
	In	*out*	*In*	*out*	*In*	*out*	*In*	*out*	*In*	*out.*		
A	0	7	7	12	12	14	14	17	17	26	– 7, 12, 14, 17	
C	7	12	12	16	16	21	21	27	27	35	- 2, 4, 1	
B	12	18	18	24	24	28	28	33	35	45	- 2, 3, 1,	2
D	18	26	26	29	29	32	33	35	45	51	25, 2	1, 1, 10
	Total		**Elapsed**		**Time**		**= 51 hours.**					

Total elapsed time is 51 hours.

Idle time for various workmen is:

P: 51 – 26 = 25 hrs.

Q: 7+(18-16)+ (26 - 24) + (51 - 29) = 33 hrs.

R: 12 + (16 – 14) + (24 – 21) + (29 – 28) + (51 – 32) = 37 hrs.

S: 14 + (21 – 17) + (28 – 27) + 51 – 35) = 35 hrs.

T: 17 + 27 – 26) = 18 hrs.

The waiting time for machines is:

A: No waiting time. The machine will finish it work by 26 th hour.

B: 12 + 35 – 33) = 14 hrs. The assembling will over by 45 th hour.

C: 7 hours. The assembling will over by 35 th hour.

D: 18 + 33 – 32) + (45 – 35) = 29 hrs. The assembling will over by 51-st hour.

Problem 6.13.

Solve the sequencing problem given below to give an optimal solution, when passing is not allowed.

Machines (Processing time in hours)

Jobs	*P*	*Q*	*R*	*S*
A	11	4	6	15
B	13	3	7	8
C	9	5	5	13
D	16	2	8	9
E	17	6	4	11

Solution

Minimum time element under machine *P* and *S* are 9 hours and 8 hours respectively. Maximum time element under machines *Q* and *R* are 6 hours and 8 hours respectively. As minimum time elements in first and last machines are > than the maximum time element in the intermediate machines, the problem can be converted into two machine, *n* jobs problem.

See that sum of the time elements in intermediate machines (*i.e.* machines Q and R is equals to 10, hence we can take first and last machines as two machines and by application of Johnson and Bellman principle, we can get the optimal solution. The optimal sequence is:

Two-machine problem is:

Job:	*A*	*B*	*C*	*D*	*E*
Machine G (Hrs)	11	13	9	16	17
Machine H (Hrs)	15	8	13	9	11

Optimal sequence:

C	A	E	D	B

Total elapsed time:

Sequence	*Machine P Time in Hrs.*		*Machine Q Time in Hrs.*		*Machine R Time in Hrs.*		*Machine S Time in Hrs.*		*Job idle Time in Hrs*	*Machine idle. Time in Hrs.*
	In	*out*	*In*	*out*	*In*	*out*	*In*	*out*		*P Q R S*
C	0	9	9	14	14	19	19	32		- 9 14 19
A	9	20	20	24	24	30	32	45	2	6 5
E	20	37	37	43	43	47	47	58	13+2	13
D	37	52	52	54	54	62	62	66		9 7 4
B	52	65	65	68	68	75	75	83		18, 9+21, 8+9 -
	Total		**Elapsed**		**Time in Hrs.**		**= 83 hrs.**			

Total elapsed time is 83 hours.

PROCESSING OF 2 - JOBS ON 'M' MACHINES

There are two methods of solving the problem. (*a*) By enumerative method and (*b*) Graphical method. Graphical method is most widely used. Let us discuss the graphical method by taking an example.

Graphical Method

This method is applicable to solve the problems involving 2 jobs to be processed on 'm' machines in the given order of machining for each job. In this method the procedure is:

(p) Represent Job 1 on X- axis and Job 2 on Y-axis. We have to layout the jobs in the order of machining showing the processing times.

(q) The **horizontal line** on the graph shows the **processing time of Job 1** and **idle time of Job 2.** Similarly, **a vertical line** on the graph shows **processing time of job 2** and **idle time of job 1.** Any inclined line shows the processing of two jobs simultaneously.

(r) Draw horizontal and vertical lines from points on X- axis and Y- axis to construct the blocks and hatch the blocks. (Pairing of same machines).

(s) Our job is to find the minimum time required to finish both the jobs in the given order of machining. Hence we have to follow inclined path, preferably a line inclined at 45 degrees (in a square the line joining the opposite coroners will be at 45 degrees).

(t) While drawing the inclined line, care must be taken to see that it will not pass through the region indicating the machining of other job. That is the inclined line should not pass through blocks constructed in step (c).

(u) After drawing the line, the total time taken is equals to Time required for processing + idle time for the job.

2. The sum of processing time + idle time for both jobs must be same.

Problem .

Use graphical method to minimize the time needed to process the following jobs on the machines as shown. For each machine find which job is to be loaded first. Calculate the total time required to process the jobs. The time given is in hours. The machining order for job 1 is *ABCDE* and takes 3, 4, 2, 6, 2 hours respectively on the machines. The order of machining for job 2 is *BCADE* and takes 5, 4, 3, 2, 6 hours respectively for processing.

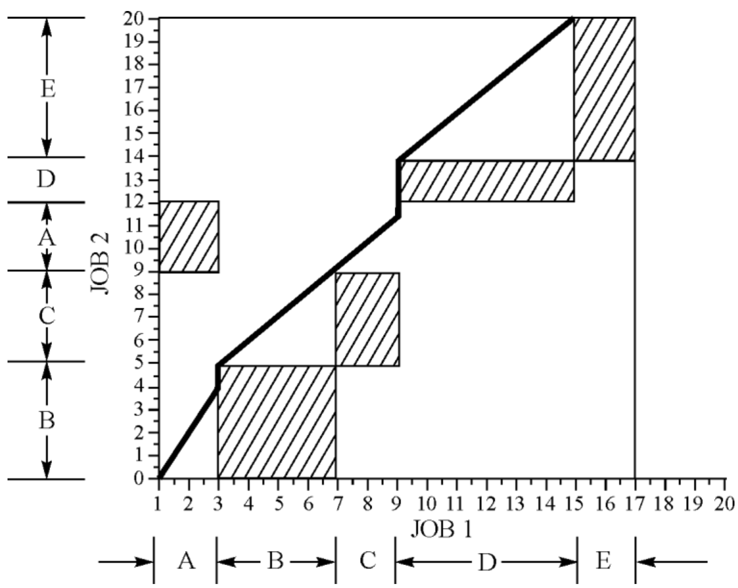
Solution

The given problem is:

Job 1	Sequence:	A	B	C	D	E
	Time in Hrs.	3	4	2	6	2
Job2	Sequence:	B	C	A	D	E
	Time in Hrs.	5	4	3	2	6

To find the sequence of jobs, *i.e*. which job is to be loaded on which machine first and then which job is to be loaded second, we have to follow the inclined line starting from the origin to the opposite corner. First let us start from origin. As Job 2 is first on machine *B* and Job 1 is first on machine *A*, job 1 is to be processed first on machine *A* and job 2 is to be processed on machine B first. If we proceed further, we see that job 2 is to be processed on machine *C* first, then comes job 2 first on *D* and job 2 first on machine *E*. Hence the optimal sequence is: (Refer figure 6.2)

Job 1 before 2 on machine A,
Job 2 before 1 on machine B,
Job 2 before 1 on machine C,
Job 2 before 1 on machine D, and
Job 2 before 1 on machine E.



The processing time for Job 1 = 17 hours processing + 5 hours idle time (Vertical distance) = 22 hours.

The processing time for Job 2 = 20 hours processing time + 2 hours idle time (horizontal distance) = 22 hours.

Both the times are same. Hence total Minimum processing time for two jobs is 22 hours.

Problem

Two jobs are to be processed on four machines *A*, *B*, *C* and *D*. The technological order for these two jobs is: Job 1 in the order *ABCD* and Job 2 in the order DBAC. The time taken for processing the jobs on machine is:

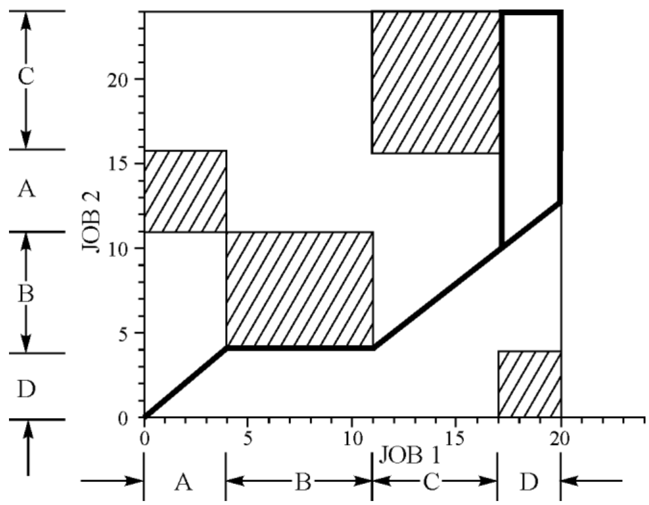
Machine:	*A*	*B*	*C*	*D*
Job 1:	4	6	7	3
Job 2:	5	7	8	4

Solution

Processing time for jobs are: Job 1 = 4 + 6 + 7 + 3 = 20 hours.

Job 2 = 5 + 7 + 8 + 4 = 24 hours.

The graph is shown in figure The line at 45 degrees is drawn from origin to opposite corner.



The total elapsed time for job 1 = Processing time + idle time (horizontal travel) = 20 + 10 = 30 hours.

The same for job 2 = Processing time + Idle time (vertical travel) = 24 + 6 = 30 hours. Both are same hence the solution. To find the sequence, let us follow inclined line.

Job 1 first on *A* and job 2 second on *A*, Job 1 first on *B* and job 2 second on *B* Job *C* first on *C* and job 2 second on *C*, Job 2 first on *D* and job 1 second on *D*.

Problem .

Find the optimal sequence of two jobs on 4 machines with the data given below:

Job 1	Order of machining:	*A*	*B*	*C*	*D*
	Time in hours:	2	3	3	4
Job 2	Order of machining:	*D*	*C*	*B*	*A*
	Time in hours:	4	3	3	2

Solution

Job 1 is scaled on *X* - axis and Job 2 is scaled on *Y* - axis. 45° line is drawn. The total elapsed time for two jobs is:

Job 1: Processing time + idle time = 12 + 2 = 14 hours.

Job 2: Processing time + idle time = 12 + 2 = 14 hours. Both are same and hence the solution;

Job 1 first on machine *A* and *B* and job 2 second on *A* and *B*. Job 2 first on *C* and job 1 second on *C*. Job 2 first on *D* and job 1 second on *D*.

Problem

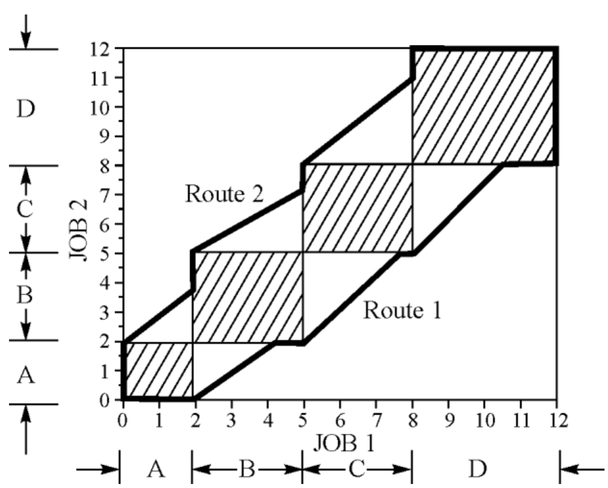
Find the sequence of job 1 and 2 on four machines for the given technological order.

Job 1.	Order of machining:	*A*	*B*	*C*	*D*
	Time in hours.	2	3	3	4
Job2.	Order of machining	*A*	*B*	*C*	*D*
	Time in hours.	2	3	3	4

Solution

From the graph figure 6.4 the total elapsed time for job 1 = 12 + 4 = 16 hours. Elapsed time for Job 2 = 12 + 4 = 16 hours.

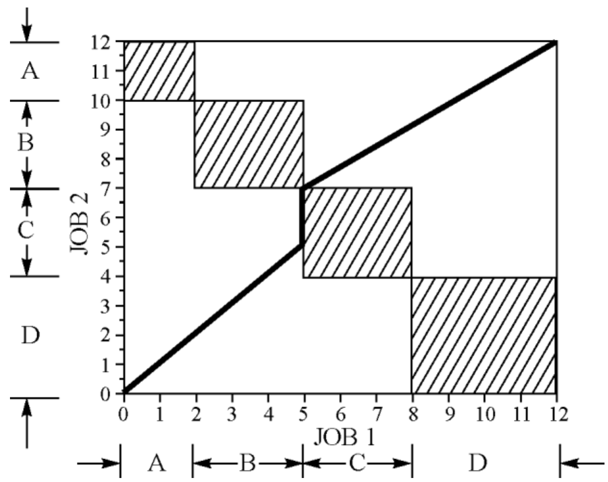
The sequence is Job 1 first on *A*, *B*, *C*, and *D* and then the job 2 is second on *A*, *B*, *C* and *D*. OR we can also do Job 2 first on *A*, *B*, *C*, *D* and job 1 second on *A*, *B*, *C*, *D*. When technological order is same this is how jobs are to be processed.



Problem

Find the optimal sequence for the given two jobs, which are to be processed on four machines in the given technological order.

Job1	Technological order:	*A*	*B*	*C*	*D*
	Time in hours.	2	3	3	4
Job2	Technological order:	*D*	*C*	*B*	*A*
	Time in hours.	2	3	3	4



Solution

(Note: Students can try these problems and see how the graph appears:

Job 1:	Technological order:	*A*	*B*	*C*	*D*
	Time in hours:	2	2	2	2
Job 2:	Technological order:	*A*	*B*	*C*	*D*
	Time in hours:	2	2	2	2

AND

Job 1	Technological order:	*A*	*B*	*C*	*D*
	Time in hours:	2	2	2	2
Job 2	Technological order:	*D*	*C*	*B*	*A*
	Time in hours.	2	2	2	2