SEQUENCING PROBLEM:

These problems arise when we are concerned with situations where there is a choice as to the order in which a number of tasks can be performed. In such cases, the effectiveness is a function of the order or sequence in which the tasks are performed. A sequencing problem could involve jobs in a manufacturing plant, aircraft waiting for landing and clearance, maintenance scheduling in a factory, programmes to be run on a computer centre, customers in a bank and so forth.

There are four different types of sequencing problems:

- Processing n jobs on two machines
- Processing n jobs on three machines.
- Processing n jobs on k machines
- Processing 2 jobs on k machines

There are some general assumptions made to solve the sequencing problems. They are:

a) The processing time on various machines is independent of the order in which different jobs are processed on them.
b) The time taken by different jobs in going from one machine to another is negligible.
c) A job once started on a machine would be performed to the point of completion uninterrupted.
d) A machine cannot process more than one job at a given point of time.
e) A job would start on a machine as soon as the job and the machine on which it is to be processed are both free.

SOLUTION TO SEQUENCING PROBLEMS:

Sequencing problems can be solved either using Gantt Charts or by applying an algorithm.

I. GANTT CHARTS

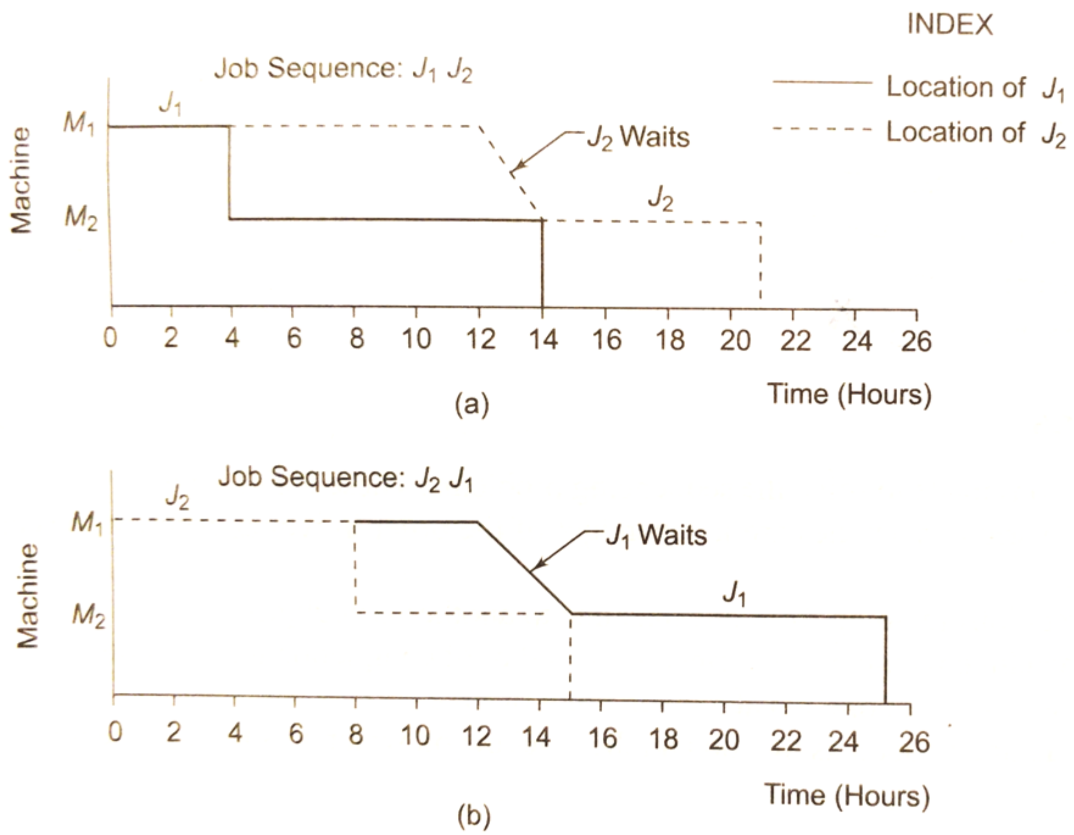
We illustrate the use of Gantt Charts with the help of the following examples:

Example 1:**Suppose that there are two jobs J1 and J2, each requiring work on two machines M1 and M2, in this order with the required processing times given as follows:**

Job	**Machine**	
	M1	M2
J1	4	10
J2	8	7

What order of performance of the jobs will involve the least time?

Solution:



For processing the two jobs, two orders are possible J1-J2 and J2-J1. Both these alternatives are evaluated in the above figure. The first part of the figure depicts the job sequences J1-J2, according to which the two jobs can be completed in 21 hours. The other sequence J2-J1 depicts that the two jobs would take 25 hours to finish.

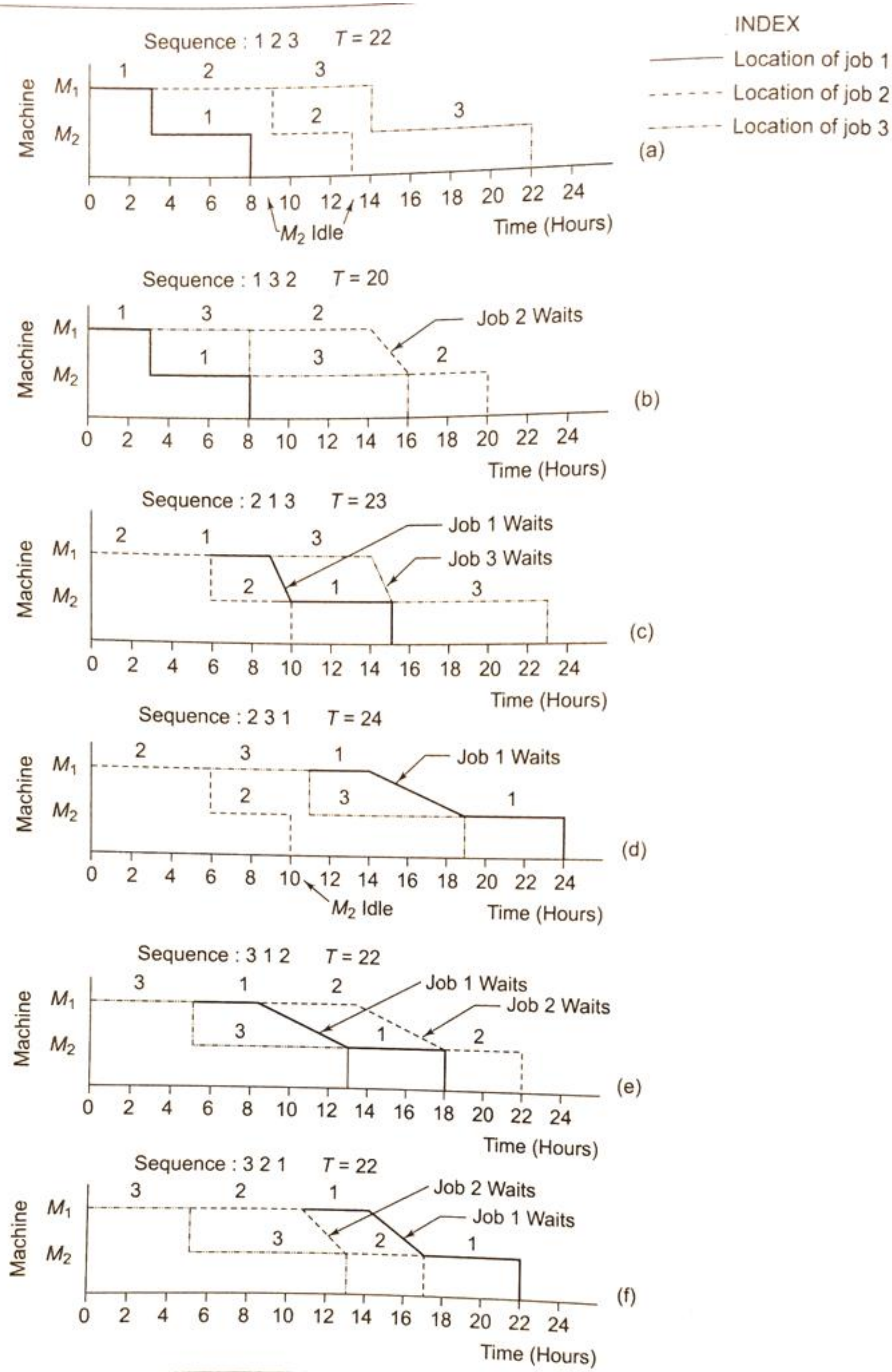
Thus, the job sequence J1-J2 is optimal, since the total elapsed time is lower of the two.

Example 2: **There are three jobs 1,2 and 3, each requiring work on two machines M1 and M2, in the order M1M2 with the required processing times given as follows:**

Job	**Machine**	
	M1	M2
1	3	5
2	6	4
3	5	8

In what sequence should the jobs be performed so that the elapsed time, T, is the least?

Solution: With three jobs to be processed on two machines, we have six possible orders namely 123, 132,213,231,312 and 321. Each of these sequences is evaluated in the chart shown below:


INDEX
Location of job 1
Location of job 2
Location of job 3
Sequence : 1 2 3 T = 22
Machine
M1
M2
M2 Idle
Time (Hours)
(a)
Sequence : 1 3 2 T = 20
Job 2 Waits
(b)
Sequence : 2 1 3 T = 23
Job 1 Waits
Job 3 Waits
(c)
Sequence : 2 3 1 T = 24
Job 1 Waits
M2 Idle
(d)
Sequence : 3 1 2 T = 22
Job 1 Waits
Job 2 Waits
(e)
Sequence : 3 2 1 T = 22
Job 2 Waits
Job 1 Waits
(f)

The sequence 132 takes minimum processing time in aggregate, that is 20 hours and hence is optimal.

As evident from the above example, the use of Gantt chart becomes disproportionately difficult as the number of jobs increases such that when there are n jobs, each of which is to be processed on two machines in the same order, there are n! possible sequences. Thus when we have 6 jobs to be processed, we need to evaluate 6!=720 possible orders in an attempt to find the optimal sequence.

Hence we need a more efficient method by which the sequence desired can be determined.

ALGORITHM FOR SOLVING SEQUENCING PROBLEMS: JHONSONS RULE

The method shall be discussed in respect of processing of n jobs through (a) two machines, (b) three machines and (c) k machines.

Processing n jobs through 2 machines:

We already have the following information:

- Only two machines are involved.
- Each job is to be processed in the order AB so that the first work would be performed on machine A and then on machine B.
- The processing time for different jobs on first machine $A_1, A_2, \ldots, A_n$ are given and so are the processing times on the second machine $B_1, B_2, \ldots, B_n$.

With the objective to determine the sequence in which the jobs may be performed so that the total time taken (i.e. the total elapsed time) is minimum, we proceed as follows.

Let A_i=Time required by job I of machine A.

B_i=Time required by job I on machine B.

T=Total elapsed time for jobs 1,2,…,n

T_i=idle time on machine B, from end of job i-1 to the start of job i.

The objective is to determine the sequence S in which n jobs may be performed so that T is minimized. The total elapsed time, T, is determined by the point of time at which the first job goes on machine A and the point of time on which the last job comes off machine B. At any moment of time machine B is either working or idle.

The total time for which machine B has to work is $\sum_{i=1}^{n} B_i$ which is determined by the given job times and has nothing to do with the sequence in which they are performed. Thus,

$$T = \sum_{i=1}^{n} B_i + \sum_{i=1}^{n} t_i$$

As evident the first term in the R.H.S of the above equation is constant, thus the main aim is to minimize the 2nd term, i.e. the total idle time on machine B. If $D_n(S)$ represents the total idle time on machine B for sequence S, one needs to find a sequence S* of jobs (1,2,…,n) such that $D_n(S^*) \leq D_n(S_0)$ for any S_0. Thus it may be stated that an optimal sequence can be obtained by using the following rule:

Job j precedes job (j+1) if $Min(A_j, B_{j+1}) < Min(A_{j+1}, B_j)$

For example, suppose there are two jobs with the following time requirements:

Job 1: Machine A :- 4 hours, Machine B:- 10 hours

Job 2: Machine A:- 8 hours, Machine B:- 7 hours

Thus A_1=4, A_2= 10, B_1=8 and B_2=7.

Here, minimum of A_1,B_2 is 4 and minimum of A_2,B_1 is 8. Since Min(A_1,B_2) < Min(A_2,B_1), it is clear that job 1 should precede job 2. If this condition was not satisfied, then job 2 would have preceded job 1.

Furthermore, if Min(A_j,B_{j+1})=Min(A_{j+1},B_j), then job j would be indifferent to job j+1; any of them could precede the other.

This rule can be extended to find the optimal sequence of a set of given jobs. Starting with any sequence S_0, the optimal sequence S* can be obtained by successive interchange of consecutive jobs by applying the above mentioned rule.

The algorithm can be stated in a stepwise manner as follows:

1) Select the smallest processing time, considering $A_1,A_2,\ldots,A_n$ and $B_1,B_2,\ldots,B_n$ together.
2)
   a. If the minimum is for A_r, i.e. for the r^{th} job on machine A, do the r^{th} job first.
   b. If the minimum is for B_s, i.e. for the s^{th} job on machine B, do the s^{th} job in the end.
   c. In case of a tie between A_r and B_s, perform the r^{th} job first and the s^{th} job last.
   d. If there is a tie between two or more timings in either of the series, select either of the jobs involved and perform them first or last, accordingly as the tie is in $A_1,A_2,\ldots,A_n$ and $B_1,B_2,\ldots,B_n$.
3) After the job(s) has/have been assigned, apply steps 1 and 2 to the reduced set of processing times obtained by deleting the machine times corresponding to the job(s) already assigned.

4) Continue in this manner until all jobs are assigned.

The sequence of the jobs to be performed obtained in this way shall be optimal, involving the least aggregate time for completion of the jobs.

***Example:* Ten jobs are required to be processed on two machines M_1 and M_2 in the order, M_1 followed by M_2. Processing times are given below. Determine the optimal sequence(s) and evaluate the total elapsed time.**

Job	**J_1**	**J_2**	**J_3**	**J_4**	**J_5**	**J_6**	**J_7**	**J_8**	**J_9**	**J_{10}**
M_1	7	8	10	3	7	4	5	8	5	6
M_2	4	2	6	6	5	7	2	6	7	6

Solution: Using Jhonson's rule, we obtain the following:

a. Minimum time is 2 hours each for job J_2 and J_7 on machine M_2. So we rank J_7 in the end, preceded by J_2. Alternatively put J_2 in the end preceded by J_7. Delete J_2 and J_7.
b. From the remaining jobs, minimum time is 3 hours for J_4 on machine M_1. So do J_4 first.
c. Now, minimum time is 4 hours for job J_1, on machine M_2 and for job J_6 on machine M_1. Therefore order J_6 at the first and J_1 at the last available places.
d. Having deleted jobs J_4, J_6,J_1,J_2 and J_7, the minimum time is 5 hours on machine M_1 in respect of job J_9. So order J_9 immediately after J_6.
e. For the remaining jobs, the least time is 6 hours for job J_{10} on machine M_1 and M_2 each, for job J_3 on machine M_2 and job J_8 on machine M_2. Thus rank J_{10} and J_8 in the end preceded or alternately succeeded by job J_3.

Thus we obtain the following alternate sequences:

S_1	**J_4 J_6 J_9 J_{10} J_3 J_8 J_5 J_1 J_2 J_7**
S_2	**J_4 J_6 J_9 J_{10} J_3 J_8 J_5 J_1 J_7 J_2**
S_3	**J_4 J_6 J_9 J_{10} J_8 J_3 J_5 J_1 J_2 J_7**
S_4	**J_4 J_6 J_9 J_{10} J_8 J_3 J_5 J_1 J_7 J_2**

The total elapsed time T for sequence S_1 is given in the below table:

Job	**Machine M_1**		**Machine M_2**	
	In	Out	In	Out
J_4	0	3	3	9
J_6	3	7	9	16
J_9	7	12	16	23
J_{10}	12	18	23	29
J_3	18	28	29	35

J_8	28	36	36	42
J_5	36	43	43	48
J_1	43	50	50	54
J_2	50	58	58	60
J_7	58	63	63	65

The total elapsed time T is 65 hours for this sequence. Other sequences can also be evaluated similarly.

Processing n jobs through 3 machines:

No general solution is available (except enumeration) for the sequencing problems of processing n jobs through 3 machines. However , we do have a method applicable under the condition that no passing of jobs is permissible (i.e. same order over each machine is maintained) and if either/both the following conditions is/are satisfied.

For three machines, A,B and C on which some jobs are to be performed in the order ABC.

Condition 1: The minimum of the times for different jobs on machine A is at least equal to the maximum of the times of different jobs on machine B.

Condition 2: The minimum of the times of various jobs on machine C is at least equal to the maximum of the times of the different jobs on machine B.

If Min A_i represents the least of the job times on machine A, Min C_i represents the least of the job times on machine C, and Max B_i indicates the largest of the job times on machine B, then either $MinA_i \geq MaxB_i$ or $MinC_i \geq MaxB_i$ or both are satisfied then the following method can be applied.

The Method:

1. Replace the given problem with an equivalent problem involving n jobs and 2 fictitious machines G and H, and define the corresponding processing times G_i and H_i as follows:
   $G_i = A_i + B_i \quad for(i = 1, 2, ..., n)$
   $H_i = B_i + C_i \quad for(i = 1, 2, ..., n)$

2. To the problem obtained in step 1, we apply the same method discussed for processing n jobs through 2 machines. The optimal sequence resulting from this shall also be optimal for the given problem.

Example: You are given the following data regarding the processing times of some jobs on three machines I, II and III. The order of processing is I-II-III. Determine the sequence that minimizes the total elapsed time (T) required to complete jobs. Also evaluate T and the idle time of II and III.

Job	Processing Time (Hours)		
	Machine		
	I	II	III
A	3	4	6
B	8	3	7
C	7	2	5
D	4	5	11
E	9	1	5
F	8	4	6
G	7	3	12

Solution: According to the given information,

$$Min\ I_i = 3$$
$$Max\ II_i = 5$$
$$Min\ III_i = 5$$

Since $Min\ III_i = Max\ II_i$, the second of the conditions specified is met. Now, the problem can be solved as follows:

Job	$G_i (= I_i + II_i)$	$H_i (= II_i + III_i)$
A	7	10
B	11	10
C	9	7
D	9	16
E	10	6
F	12	10
G	10	15

According to this, there are two optimal sequences. They are:

S_1 : A D G B F C E

S_2 : A D G F B C E

We now evaluate S_1 for the value of T as shown in the below table:

Job	Machine I		Machine II		Machine III	
	In	Out	In	Out	In	Out
A	0	3	3	7	7	13
B	3	7	7	12	13	24
C	7	14	14	17	24	36
D	14	22	22	25	36	43
E	22	30	30	34	43	49
F	30	37	37	39	49	54
G	37	46	46	47	54	59

The idle time for various machines is as follows:

Machine II: 3 + 2 + 5 + 5 + 3 + 7 + 12 = 37 hours

(0-3) (12-14) (17-22) (25-30) (34-37) (39-46) (47-59)

Machine III: 7 hours (0-7)

PRACTICE QUESTIONS

1. **A company plans to fill six positions. Since the positions are known to vary considerably with respect to skill and responsibility, different types of aptitude tests and interviews are required for each. While the aptitude tests are conducted by people from the clerical positions, the job interviews are held by the personnel from the management cadre. The job interviews immediately follow the aptitude test. The time required (in minutes) by each of the positions is given below:**

Position	Aptitude Test	Job Interview
P_1	140	70
P_2	180	120
P_3	150	110
P_4	200	80
P_5	170	100
P_6	100	90

If it is desired to minimize the waiting time of the management personnel, in what order the position filling be handled?

2. **A firm works 40 hours a week and has a capacity of overtime work to the extent of 20 hours in a week. It has received seven orders to be processed on three machines A, B and C , in the order of A,B,C to be delivered in a week's time from now. The process time (in hours) are recorded in the given table.**

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

The manager, who, in fairness, insists on performing jobs in the sequence in which they are received, is refusing to accept an eighth order, which requires 7,2 and 5 hours respectively on A,B and C machines, because, according to him, the eight jobs would require a total of 61 hours for processing, which exceeds the firms capacity. Advise him.

3. **Find the sequence that minimizes the total elapsed time required (T) in completing the following jobs. Each job is processed in the order ABC. Also calculate T.**

Job	Machine A	Machine B	Machine C
1	10	6	8
2	8	4	7
3	12	6	5
4	6	5	9
5	9	3	10
6	11	4	6
7	9	2	5