Quiz-1 OS-2

Name: Naitik Malav
Roll No: CS19BTECH11026

Ans4. The maximum number of possible schedules are n! (n factorial).
Reason:
Because it takes 1 process at a time and completes it.
So 1st process can be executed in n ways,
2nd in (n-1) ways
3rd in (n-2) ways and so on. So n process can be executed in n! (n factorial) ways.

Ans5.
a) The task may be delayed by upto time t and still meets its deadline.
b) The task must be executed right away. Otherwise it will not meet it’s deadline.
c) The task cannot meets its deadline.

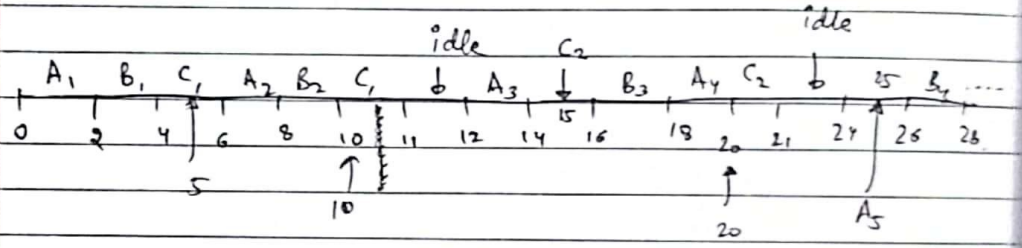
Ans 5 (d)

Task	Period	Execution Time
A	6	2
B	8	2
C	12	3.

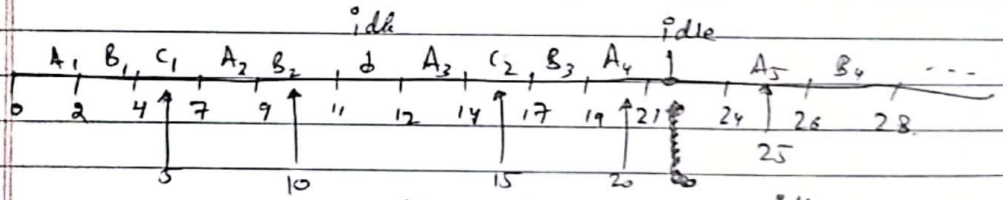
$u_1 = \frac{2}{6} = 33\%$ $u_2 = \frac{2}{8} = 25\%$ $u_3 = \frac{3}{12}$ 25%

$$\sum_i u_i = 83\% < 100\%.$$

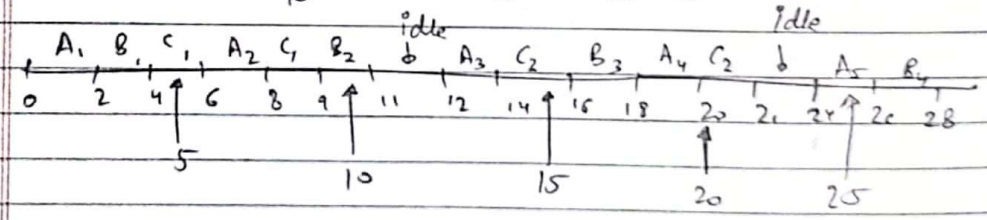
RMS:



EDF:



LLF:



In all classes the pattern repeats every 24ms.

Ans1.

Ans 1.

Process	Arrival Time	CPU Burst
P_1	0	3
P_2	1	1
P_3	2	3
P_4	3	3

avg. waiting time = 1 ms.

Gant chart. (for below assumption)

t:	0	1	2	3	4	5	6	7	8
	P_1	P_2	P_1	P_1	P_4	P_3	P_3	P_3	
Assumption: remaining time →	(2)	(0)	(1)	(0)	(0)	(2)	(1)	(0)	

if z = 1, then avg. waiting time is →

$$1 = \frac{(1 + 0 + 2 + 1)}{4} = 1$$

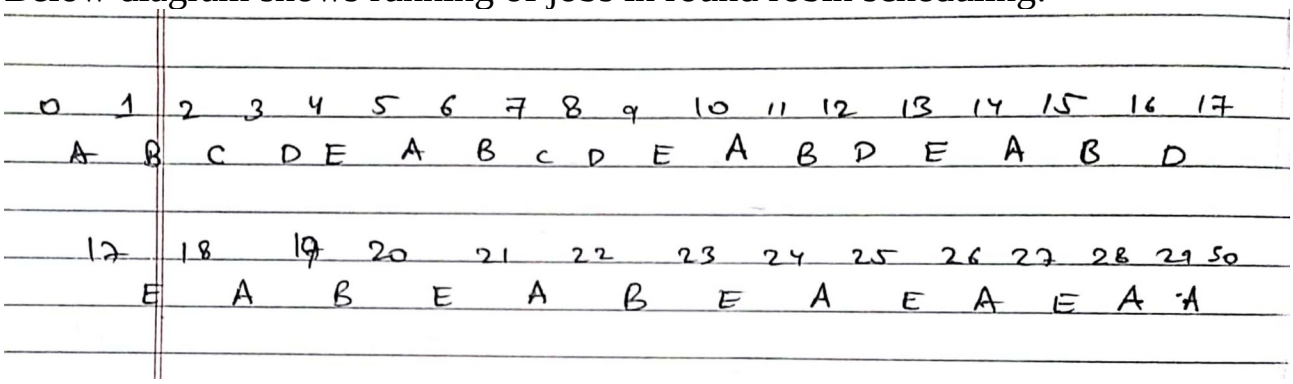
So z = 1 ag

our assumption is correct.

Ans 3.

a) Suppose quantum time is 1ms. So each job is allowed to run for 1ms after which it is preempted.

Below diagram shows running of jobs in round robin scheduling:



Process	Turn Around time(in ms)
A	30
B	23
C	8
D	17
E	28

So mean process turn around time is : (30+23+8+17+28)/5 = <u>21.2 ms</u>

b) Each job run to completion without being preempted.

B has highest priority so it will run first, then followed by E, A, C and finally D.

No preemption so -

1. B completes after 6ms
2. E after 6ms of waiting and 8ms of processing, so total 14ms.
3. A after 14ms of waiting for B and E, and 10ms of processing, so total 24ms.
4. C after 24ms of waiting for B, E and A, and 2ms of processing, so total 26ms.
5. D after 26ms of waiting for C, B, E and A, and 4ms of processing, so total 30ms.

So avg is (6+14+24+26+30)/5 = 100/5 = <u>20ms</u>.

c) Process runs to completion in order of arrival.

1. A completes after 10ms.
2. B after (10+6) = 16ms.
3. C after (16+2) = 18ms.
4. D after (18+4) = 22ms.
5. E after (22+8) = 30ms.

So avg is (10+16+18+22+30)/5 = 96/5 = 19.2ms

d) Process will run in order C, D, B, E, A

1. C completes after 2ms.
2. D completes after (2+4) = 6ms.
3. B completes after (6+6) = 12ms.
4. E completes after (12+ 8) = 20ms.
5. A completes after (20+10) = 30ms.

So avg is : (2+6+12+20+30)/5 = 70/5 = 14ms.

2.

a) In case of b<<a then priorities of process in holding queue will increase very fast as compared to running queue. Due to this the priority of the process that are currently in the holding queue will increase very fast and the self round robin will behave as a normal round robin because hodling queue willreach the level of active queue very fast and there is no use in maintaining two diff queues. As soon as a process is added to the holding queue it will move to the active queue and start competing for CPU time.

b) If a==b then it will behave like first some first serve, the priority of all process are incremented with the same value over time. As soon as old process are terminated, process added at similar time will be added to the active queue. The process that has been longest in the holding queue will reach the level of active queue first and compete for CPU time.