

①

Name: M. Usman Khan

Roll No: 2115

Paper: Operating System

Date: - 15-june-2021

Question 1) a)

Highest Priority → 

FCFS
------

 → Queue 1

→ 

Round Robin
-------------

 → Queue 2

Lowest Priority → 

SJF
-----

 → Queue 3

Question 1) b)

highest → 

P3	P6	P7	P9
----	----	----	----

 → Queue 1

→ 

P1	P5	P10	P12
----	----	-----	-----

 → Queue 2

→ 

P2	P4	P8	P11
----	----	----	-----

 → Queue 3

(2)

Question 1) c)

Queue 1: 

P3	P6	P7	P9
----	----	----	----

 FCFS  
0 26 57 86 98

Queue 2: 

P1	P5	P10	P12	P1	P5	P10
----	----	-----	-----	----	----	-----

 →  
98 105 112 119 126 133 140 147

Round Robin → 

P12	P1	P5	P10	P12	P5	P10	P5	P10	P5
-----	----	----	-----	-----	----	-----	----	-----	----

  
147 154 161 168 175 180 187 194 201 207

Queue 3: 

P11	P2	P4	P8
-----	----	----	----

 SJF  
207 218 231 246 270

Question 1) d)

Queue 1 FCFS

Process	Burst Time	W.T	T.A.T	C.T
P3	26	0	26	26
P6	31	26	57	57
P7	29	57	86	86
P9	12	86	98	98

$$\text{Avg WT} = \frac{0 + 26 + 57 + 86}{4} = \frac{169}{4} = 42.25 \text{ ms}$$

$$= 2678 \text{ ms}$$

$$\text{Avg T.A.T} = \frac{26 + 57 + 86 + 98}{4} = \frac{267}{4}$$

$$= 66.75 \text{ ms}$$

Queue 2)

Round Robin

(3)

Processes	AT	BT	CT	T.AT	WT
P1	98	21	161	63	42
P5	98	36	207	109	73
P10	98	33	206	108	85
P12	98	19	180	82	63

$$\text{Avg WT} = \frac{42 + \cancel{73} + \cancel{75} + 63}{4} = \frac{253}{4}$$

$$= 63.25 \text{ ms}$$

$$\text{Avg T.AT} = \frac{63 + 109 + 108 + 82}{4} = \frac{362}{4}$$

$$= 90.5 \text{ ms}$$

Queue 3)

STF

Process	A.T	B.T	C.T	T.AT	W.T
P2	207	13	231	24	11
P4	207	15	246	39	24
P8	207	24	270	63	39
P11	207	11	218	11	0

$$\text{Avg W.T} = \frac{11 + 24 + 39}{4} = \frac{74}{4}$$

$$= 18.5 \text{ ms}$$



(4)

$$\begin{aligned} \text{By } T.A.T &= \frac{24 + 39 + 63 + 11}{4} = \frac{137}{4} \\ &= 34.25 \text{ ms} \end{aligned}$$

(5)

Q2) a) What is deadlock?

Ans) Deadlock is a situation where a set of processes are blocked because each process is holding a resource and waiting for another resource acquired by some other process. if process A is waiting for a resource that process B has and process B is waiting for a resource that process A has.

Q2) b) Why a deadlock occurs?

Ans) Deadlock occurs when any process enters a waiting state because another waiting process is holding the demanded resources. They generally occur in non-preemptable resources. if It also occurs when let's say process A ~~requests~~ is waiting for a resource that process B has and process B is waiting for a resource that process A has. Therefore a deadlock occurs.

Q2) c) Explain the condition that

Ans) There are following 4 necessary conditions for occurrence of deadlock.

(2)

## 1- Mutual Exclusion :-

By this condition,

- There must exist at least one resource in the system which can be used by only one process at a time
- If there exists no such resource, then deadlock will never occur
- Printer is an example of a resource that can be used by only one process at a time

## 2- Hold and Wait :-

By this condition,

- There must exist a process which holds some resources and waits for another resource held by some other process

## 3- No Preemption :-

By this condition,

- Once the resource has been allocated in the process it can not be pre-empted
- It means resource cannot be snatched forcefully from one process, to another and given
- The process must release the resource voluntarily by itself

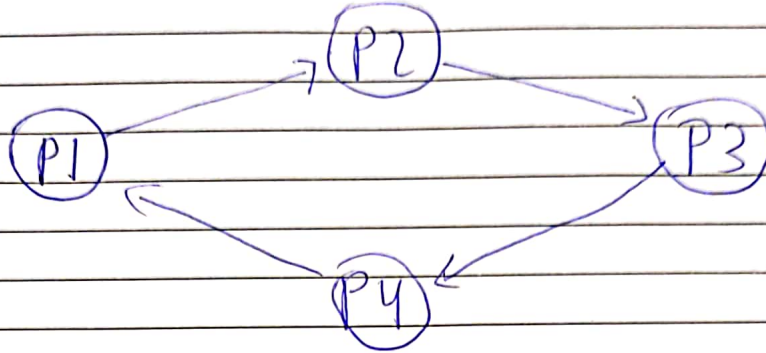
## 4- Circular Wait :-

By this condition,



(7)

- All the processes must wait for the resource in a cyclic manner where the last process waits for the resource held by the first process.



here :

Process P1 waits for a resource held by P2

Process P2 waits for a resource held by P3

Process P3 waits for a resource held by P4

Process P4 waits for a resource held by P1

Note :-

- All these 4 conditions must simultaneously for the occurrence of deadlock

- if any of these conditions fail, system is deadlock free

(8)

Q2) a) How these conditions can be prevented? and at what cost?

For deadlock to occur, 4 necessary conditions must hold. If we manage to overcome any of them, we can prevent deadlock.

1- Mutual Exclusion:

if a resource is not assigned separately to a process, then we will not have deadlock. but there are some resources that cannot be assigned to more than one process simultaneously. so to avoid it, only a few processes should request for such kind of resources.

2) Hold & wait:

We may overcome hold & wait problem if process request for all the resources that it needs before starting execution. if all the requested resources are available then the process will be executed otherwise process will wait for those resources.

3) Non-Preemption:

if a Process that is holding



(9)

Some resources request another resource that cannot be allocated to it, then it will release its resources for use by other processes. Other processes that could have waited for those resources will use these preempted resources, and after completion return them to system. Now the process that sacrificed its resources for other processes will start again and will have all the resources available.

4 - Circular wait :-

To avoid this, only one process can use only resource at any moment. if it needs another, then it should release the first resource.

Cost :-

each condition has a different cost. i.e.

Mutual Exclusion :-

~~not all resources~~  
There are some resources which

cannot be assigned to more than 1 process simultaneously

Held by unit :-

This is a time consuming & wastes resources but deadlock can be prevented

Non-Preemption :-

This can be applied to resources but cannot be used for resources like tape or Printer

Circular wait :-

This cannot be used if we want to print a huge file.