S.No.:01SK_ABCDEFG_20052025

MADE EASY
Leading Institute for IES, GATE & PSUs
Delhi | Bhopal | Hyderabad | Jaipur | Pune | Kolkata
Web: www.madeeasy.in | E-mail: info@madeeasy.in | Ph: 011-45124612

Operating System

COMPUTER SCIENCE & IT

Date of Test : 20/05/2025

ANSWER KEY ➤

1.	(d)	7.	(d)	13.	(b)	19.	(d)	25.	(d)
2.	(d)	8.	(b)	14.	(c)	20.	(d)	26.	(b)
3.	(c)	9.	(a)	15.	(b)	21.	(c)	27.	(d)
4.	(c)	10.	(b)	16.	(c)	22.	(b)	28.	(c)
5.	(b)	11.	(c)	17.	(b)	23.	(c)	29.	(c)
6.	(a)	12.	(d)	18.	(d)	24.	(a)	30.	(d)

DETAILED EXPLANATIONS

1. (d)

1. Since, each process has its own address space, it needs to involve the Kernel when dealing with other process address space.
2. A software interrupt is required to switch between the two modes.
3. In both synchronous and asynchronous I/O an ISR is invoked after completion of the I/O.
4. Statement is correct.

2. (d)

(a) Round robin works on time quantum, after certain period of time every process gets the CPU unit for its completion, hence it's most suitable.

(b) Since OS is multiuser and multiprocessing, hence security is the primary concern so that user processes and Kernel processes can be isolated.
Hence two modes are required.

(c) When CPU temperature is too high, the BIOS initiate an interrupt. OS given top priority to this interrupt.

(d) Address translation table need to be changed when switching context from process A to process B.

3. (c)

1. CPU senses interrupt request line after every instruction.
2. Nearest cylinder next disk scheduling strategy gives the best throughput but the only problem is it can lead to starvation.
3. Using large file block size in a fixed block size file system leads to better disk throughput but poor disk space utilization.

4. (c)

If mutual exclusion is satisfied on shared resources, then RACE condition does not arises.
If the processes are of cooperative processes, then they require synchronization.

5. (b)

- When, CPU utilization is low and disk utilization is very high, the situation is called thrashing.
- In (3), CPU utilization as well as disk utilization both are low, hence, it can be said that, there is a need to increase multiprogramming.

6. (a)

$$\begin{aligned}\text{Page Size} &= 8\text{ K}\\ \text{Offset bits} &= 13\\ \text{Virtual Address} &= 64\text{ bits}\\ \text{Remaining bits} &= 64 - 13 = 51\text{ bits}\\ \text{Number of sets} &= \frac{256}{4} = 64 = 6\text{ bits}\\ \text{Tag bits} &= 51 - 6 = 45\text{ bits}\end{aligned}$$

7. (d)

Both (a) and (c) are correct.

If there are too many programs running in the system then it is a cause of thrasing.

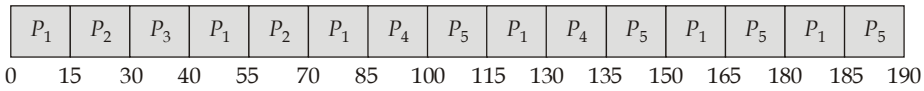
8. (b)

$$\begin{aligned} \text{Resources} &= 13 \\ \text{Process} &= n \\ \text{Requirement} &= \text{Max} = 5 \\ [n-1] \times 4 + 5 &= 13 \quad \text{[Ensuring atleast one process has 5 resources, i.e. no deadlock]} \\ 4n - 4 + 5 &= 13 \\ 4n &= 13 + (-1) \\ n &= 3 \end{aligned}$$

If value of $n = 3$ then there is no deadlock.
So minimum value of n is 4 which lead to deadlock.

9. (a)

Gantt chart:



Processes	Arrival Time	Burst Time	Turn Around Time
P_1	0	80	185
P_2	10	30	60
P_3	10	10	30
P_4	80	20	55
P_5	85	50	105
			Average = 87 msec

10. (b)

$$\begin{aligned} 50 &= 0.3\,(P * 400 + (1 - P) * 125) + 0.7 * 10 \\ \Rightarrow \quad 50 &= 0.3(400\,P + 125 - 125\,P) + 7 \\ \Rightarrow \quad 43 &= 0.3\,(275\,P + 125) \\ \Rightarrow \quad 43 &= 82.5\,P + 37.5 \\ \Rightarrow \quad 43 - 37.5 &= 82.5\,P \\ \Rightarrow \quad 5.5 &= 82.5\,P \\ \Rightarrow \quad P &= \frac{5.5}{82.5} = 0.0667 \end{aligned}$$

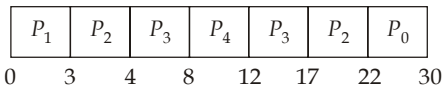
11. (c)

Many to one model maps many user threads to one Kernel thread.

12. (d)

There is a bit in PSW which will be change when mode is change from Kernel to user or user to Kernel.
Process identification is a part of process control block.

13. (b)



Gantt Chart

Waiting time = Turn around time – Arrival time

Process	Waiting Time
P_0	22
P_1	0
P_2	14
P_3	4
P_4	0

$$\text{Average waiting time} = \frac{22+0+14+4+0}{5} = 8$$

14. (c)

(a) Wait (B), wait (A), wait (A), wait (B).
If X_1 is executed in process P and then process is preempted and X_3 is executed there is deadlock condition.

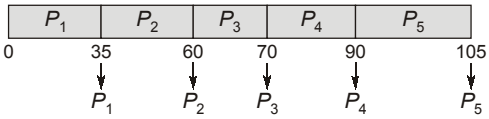
(b) Wait (A), wait (B), wait (B), wait (A)
There also deadlock may occurs.

(c) Wait (A), wait (B), wait (A), wait (B)
This is the correct implementation.

15. (b)

If, we remove the lock while acquiring the fork. It may lead to deadlock, if all process execute (i) statement before any philosopher has execute (ii) statement.
Removal of (iii) and (iv) will not affect the code, since no conflict can occur doing the V operation on forks.

16. (c)

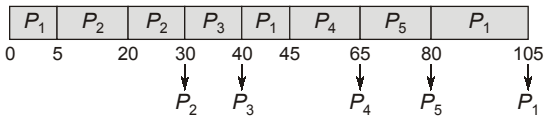


Calculating TAT:

P_1 : 35 – 0 = 35
P_2 : 60 – 5 = 55
P_3 : 70 – 20 = 50
P_4 : 90 – 45 = 45
P_5 : 105 – 65 = 40

$$\text{Avg TAT} = \frac{35+55+50+45+40}{5} = 45$$

SRT:



Calculating TAT:

P_1 : 105 – 0 = 105
P_2 : 30 – 5 = 25
P_3 : 40 – 20 = 20
P_4 : 65 – 45 = 20
P_5 : 80 – 65 = 15

$$\text{Avg TAT} = \frac{105+25+20+20+15}{5} = 37$$

Difference = 45 – 37 = 8

17. (b)

Page size 1 K = 1024 word

Page	Frame number
0	0 - 1023
1	1024 - 2047
2	2048 - 3071
3	3072 - 4095
4	4096 - 5119
5	5120 - 6143
6	6144 - 7167
7	7168 - 8191

Page 0, 2, 5, 7 are already is memory so for these pages there is no page fault.

(a) Pages 0, 2, 4, 5 are needed for page 4 there is page fault.
(b) Pages 0, 2, 5, 7 are needed so no page fault.
(c) Pages 1, 3, 4, 6 so page fault.
(d) Pages 0, 1, 5, 7 for 1 there is page fault.

18. (d)

S_1: Translation look aside buffer need not to be saved.
S_2: Program counter, stack counter and general purpose register must be saved when context switch.
S_3: System calls are usually invoked by using a software interrupt.

19. (d)

I. Dirty bit is used to check that page is modified or not, to avoid unnecessary write.
II. Belady's anomaly indicates page fault rate may increase on increasing the number of allocated frames.
III. Thrashing can not be avoided by swapping.

20. (d)

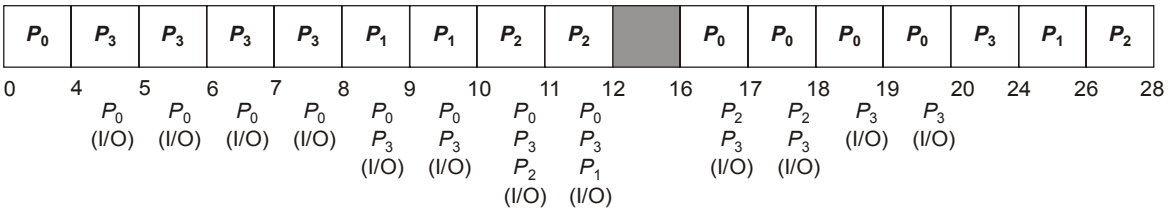
	X	Y	Z	W
P_0	2	2	2	2
P_1	3	2	0	0
P_2	0	3	2	4
P_3	2	5	0	2
P_4	2	0	0	1

Since available is a 0 0 b, let's suppose a takes value 2 and b takes the value 1.
Available = 2 0 0 1
$P_4 \rightarrow$ Complete $\rightarrow$ Avail = (0000 + 6214) = 6214
$P_1 \rightarrow$ Complete $\rightarrow$ Avail = (6214) – (3200) = (3014) + (3512) = (6526)
$P_0 \rightarrow$ Complete $\rightarrow$ Avail = (6526) – (2222) = (4304) + (3242) = (7546)
$P_2 \rightarrow$ Complete $\rightarrow$ Avail = (7546) – (0324) = (7222) + (2775) = (9, 9, 9, 7)
$P_3 \rightarrow$ Complete $\rightarrow$ Avail = (9997) – (2502) = 7495
Hence, the system is in a safe state will value of a as 2 and value of b as 1.

21. (c)

Process	Burst Time	CPU	I/O	CPU	CT	TAT
P_0	20	4	12	4	20	20
P_1	10	2	6	2	26	26
P_2	10	2	6	2	28	28
P_3	20	4	12	4	24	24
						Average TAT = 98/4 = 24.5

Gantt Chart



22. (b)

- Rotational latency $\Rightarrow$ 6000 rotation $\rightarrow$ 60 sec
  1 rotation $\Rightarrow$ 1/100 sec

$$\text{Rotational latency} \Rightarrow \frac{1}{2}\times\frac{1}{100} = \frac{1}{200}\text{sec} = 0.005\,\text{sec}$$

$$\text{Transfer time = 64 KB} \Rightarrow \frac{1}{100}\text{sec}$$

$$1\text{ KB} \Rightarrow \frac{1}{6400}\text{sec} = 0.000156\text{ sec}$$

$$\text{Data transfer rate} \Rightarrow \frac{1}{100}\text{sec} \rightarrow 64\text{ KB}$$

$$1\text{ sec} \rightarrow 64\text{ KB} \times 100$$
$$\rightarrow 6400\text{ KBPs}$$

- Time required to read 800 random sectors
  Total time required = [Seek time + RL + TT (1 sector)] × 800

$= (0.005 + 0.005 + 0.000156) \times 800$
$= 8.12$ sec

- Total time = Seek time + Rotational latency + Transfer time
  $= 5$ msec $+ 0.005$ sec $+ 0.1248$ sec
  $= 134.8$ msec

23. (c)

Initial value of semaphore = x

3rd step: 22 V operations = $x + 22$

2nd step: 12 P operations = $x + 22 - 12 = x + 10$

1st step: 3 V operations = $x + 10 + 3 = x + 13$

As per question, $x + 13 = 20$

$\Rightarrow \quad x = 7$

24. (a)

By using LRU policy:

9	8	7	9	3	0	2	9	8	3	9	2	0	9
					0	0	0	0	0	0	0	0	0
				3	3	3	3	3	3	3	3	3	3
		7	7	7	7	7	7	8	8	8	8	8	8
	8	8	8	8	8	2	2	2	2	2	2	2	2
9	9	9	9	9	9	9	9	9	9	9	9	9	9
F	F	F	H	F	F	F	H	F	H	H	H	H	H

7 faults

By using FIFO policy:

9	8	7	9	3	0	2	9	8	3	9	2	0	9
					0	0	0	0	0	0	0	0	0
				3	3	3	3	3	3	3	3	3	3
		7	7	7	7	7	7	8	8	8	8	8	8
	8	8	8	8	8	8	9	9	9	9	9	9	9
9	9	9	9	9	9	2	2	2	2	2	2	2	2
F	F	F	H	F	F	F	F	F	H	H	H	H	H

8 faults

So, [LRU – FIFO] page fault: 7 + 8 = 15

25. (d)

- The total size of address space in a virtual memory system is limited by the available secondary storage.
- Best fit technique can also suffer from fragmentation.
- Locality of reference implies that the page reference being made by a process is likely to be the page used in the previous page reference.
- In a system with virtual memory context switch includes extra overhead in switching of address space.

26. (b)

Direct block addressing = $16 * 256 \Rightarrow 4$ KB

Single indirect block addressing = $\left[\frac{256}{8}\right] * 256\text{B}$

$= 2^5 * 2^8$ B

$= 2^{13}\ B \Rightarrow 8\ KB$

1 doubly indirect block addressing $\Rightarrow \left[\frac{256}{8}\right]^2 * 256\ B$

$\Rightarrow (2^5)^2 * 2^8\ B$
$\Rightarrow 2^{10} * 2^8\ B$
$\Rightarrow 256\ KB$

1 triple indirect block addressing $\Rightarrow \left[\frac{256}{8}\right]^3 * 256\ B$

$\Rightarrow (2^5)^3 * 2^8\ B$
$\Rightarrow 2^{15} * 2^8\ B$
$\Rightarrow 2^{23}\ B$
$\Rightarrow 8\ MB$

27. (d)

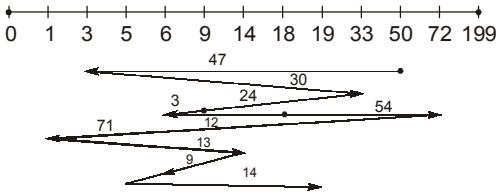
The output is 'TGE'. So, to print 'T', we must give a value of 1 to semaphore b and should block rest three processes.

Now, process 3, after printing T, will give signal to semaphore a, which will wake up process 1 and will print 'G' and given signal to semaphore 'b' and 'c'. On giving signal to semaphore 'c', process '2' will get awake. But 'a' should not be printed in the output hence 'c' should be given value '–1'.

Process 4 will also awake after process 3 on signal 'a', but it will again be blocked by wait (b).

28. (c)

- **FCFS**



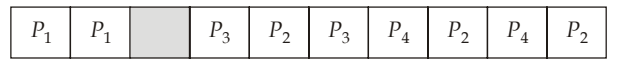
Total time = 47+30+27+12+54+71+13+9+14 = 277 msec

29. (c)

P.T.E. = Frame bits + Others
= 11 frame bits + 1 (valid bit) = 12 bits
Page size = 512 byte
So, Page offset bit = 9 bit
So, Physical address = Frame bit + Offset bit
= 11 + 9 = 20 bit
So, Physical address space size = 2^{20} bytes

30. (d)

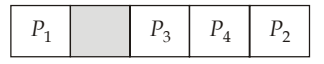
Round Robin with quantum 6



$P_1\ P_1\ P_3\ P_2\ P_3\ P_4\ P_2\ P_4\ P_2$

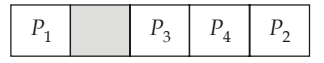
Processes	Arrival Time	Completion Time	Turn Around Time
1	0	10	10
2	18	44	26
3	12	26	14
4	20	41	21
			Average = $\frac{71}{4}$ = 17.75

SJF



Processes	Arrival Time	Completion Time	Turn Around Time
1	0	10	10
2	18	44	26
3	12	20	8
4	20	29	9
			Average = $\frac{53}{4}$ = 13.25

SRTF



Processes	Arrival Time	Completion Time	Turn Around Time
1	0	10	10
2	18	44	26
3	12	20	8
4	20	29	9
			Average = $\frac{53}{4}$ = 13.25

■■■■