S.No. : 01PT_CS_A+B_080522

MADE EASY
India's Best Institute for IES, GATE & PSUs

Delhi | Bhopal | Hyderabad | Jaipur | Lucknow | Pune | Bhubaneswar | Kolkata | Patna

Web: www.madeeasy.in | **E-mail:** info@madeeasy.in | **Ph:** 011-45124612

OPERATING SYSTEM

COMPUTER SCIENCE & IT

Date of Test : 10/05/2022

ANSWER KEY ➤

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

DETAILED EXPLANATIONS

1. (c)

Reference string 0, 1, 3, 1, 2, 4, 7, 3, 0, 2, 1, 3

0	2	2	2	3	3	3	1	1
1	1	1	7	7	7	2	2	2
3	3	4	4	4	0	0	0	3
3 fault	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)

Total 11 page fault occurrences.

2. (a)

Total 8 process and each process required maximum of 3 instances of 'R'.

For deadlock $R <= (3 - 1) \times 8$

$R <= 2 \times 8$

$R = 16$

Maximum value of R is 16 if R is 17 there is no deadlock.

3. (c)

First fit allocates the first free hole that fulfill the requirement.

Best fit allocates the smallest hole that fulfill the requirement.

Next fit allocates the first free hole from the last allocation.

4. (d)

Processes and Kernel level thread can run in parallel in multiprocessor system Kernel level thread treated independently.

Reading the clock of system can be done in user mode but accessing the I/O devices needs privileged instruction.

So all the statement is correct.

5. (c)

Give the sequence of holes

10 KB, 4 KB, 17 KB, 18 KB, 7 KB, 9 KB, 12 KB, 15 KB, 20 KB

First fit

The segment of size 12KB will occupy the first hole in which it can accommodate.

Hence, 17 KB hole is occupied.

Best fit

The hole in which least space is left after filling the segment is selected.

Hence, 12 KB hole is occupied.

Worst fit

The hole with highest size is selected.

Hence, 20 KB hole is occupied.

6. **(a)**

Long term scheduler create a new process and loads into memory by making a transition from new to ready.

Long term schedule controls the degree of multiprogramming.

So option (a) is correct.

7. **(c)**

Access time: total time needed to access the data

Access time = seek time + rotational latency + data transfer time

Seek time: Time taken to move the head to the correct cylinder that contains desired sector.

Rotational latency: Time taken to move the head to the desired sector within the cylinder.

Data transfer time : Time taken to transfer the actual data.

8. **(b)**

If the time quantum of Round Robin is greater then Round Robin will act same like FSFS scheduling algorithm.

9. **(d)**

New process in this algorithm will always ran immediately and older process will sit in the ready queue until the most recent process finishes. This is essentially last in first out algorithm.

10. **(d)**

Compiler need not be part of OS, included in standard libraries, which can be accessed by user directly.

CPU scheduling, Page replacement and virtual memory are part of operating system, which need to be privileged.

11. **(b)**

S_1 : The essential content in page table entry is frame number not virtual page number.

S_2 : Segmentation is suffers from external fragmentation.

S_3 : Paging can suffer from internal fragmentation.

Only S_2 and S_3 is correct.

12. **(d)**

S_1 : Context switching time in uses level threads will be less compared to Kernel level threads.

S_2 : Threads can share the code segment.

S_3 : Kernel level threads are best suitable for I/O bound processes.

13. **(c)**

(i) For segment 0, length = Logical address 99 < 124
∴ Physical address = 330 + 99 = 429

(ii) For segment 2, length = Logical address 78 < 99
∴ Physical address = 111 + 78 = 189

(iii) For segment 1, length = Logical address 265 < 211
∴ Trap

(iv) For segment 3, length = Logical address 222 < 302
∴ Physical address = 498 + 222 = 720

(v) For segment 0, length = Logical address 111 < 124
∴ Physical address = 330 + 111 = 441

14. (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.

15. (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.

16. (b)

Value of id can be 0 or 1 at a time and only one process can enter into critical section at a time so mutual exclusion is satisfied.
There are strict alteration so progress is not satisfied.

17. (a)

Race condition is occur if more than one process entering into critical section at the same time. If mutual exclusion is satisfies race condition will not occur, if only one process is trying to update the shared variable race condition is not occur, it if possible if more than one process trying to update at same time.

18. (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.

19. (b)

$$r = 0, s = 0, t = 1$$

First process 3 will be execute only because $r = 0$ and $s = 0$ so process 1 and process 2 can not execute.
Process 3 will print ABA and after signal (r), r value become 1 so process 2 will execute and print ABC and last process 1 will execute and print AB.
So finally "ABABCAB" will be printed.
other options may result "ABABCAB" but not always.

20. (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.

21. (a)

Mutual exclusion will always hold for these processes since only 1 process can enter the critical section at a time. There may be deadlock.

22. (d)

FCFS:

A	B	C	D	E

0 3 7 10 15 22

Process Name	Arrival Time	Burst Time	Completion Time	TAT Time	Waiting Time
A	0	3	3	3	0
B	3	4	7	4	0
C	7	3	10	3	0
D	9	5	15	6	1
E	12	7	22	10	3

4/5 = 0.8

RR (*T.Q.* = 3):

A	B	B	C	D	E	D	E

0 3 6 7 10 13 16 18 22

Process Name	Arrival Time	Burst Time	Completion Time	TAT Time	Waiting Time
A	0	3	3	3	0
B	3	4	7	4	0
C	7	3	10	3	0
D	9	5	18	9	4
E	10	7	22	12	5

9/5 = 1.8

SRTF:

A	B	C	C	D	D	E

0 3 7 9 10 12 15 22

Process Name	Arrival Time	Burst Time	Completion Time	TAT Time	Waiting Time
A	0	3	3	3	0
B	3	4	7	4	0
C	7	3	10	3	0
D	9	5	15	6	1
E	10	7	22	12	3

4/5 = 0.8

Since FCFS time = SRTF time < RR time.

23. **(b)**

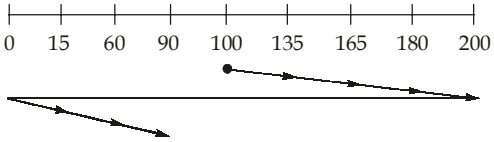
Each page size = 1 KB

Size of array [A] = 1000 × 4= 4000 B

⇒ Number of pages required for A is 4 to hold 4000 bytes

TLB misses = 4

24. **(a)**



Total seek time = (135 – 100) + (165 – 135) + (180 – 165) + (200 – 180) + 10 + (15 – 0) + (60 – 15) + (90 – 60)

= 35 + 30 + 15 + 20 + 10 + 15 + 45 + 30 = 200 msec

25. **(d)**

I. Dynamic linking decreases memory space utilization since the shared libraries gets linked during the execution and there is no need to store these initially.

II. Dynamic linking increases compile time and increases program execution time.

III. Most of the operating system uses dynamic linking.

26. **(c)**

Minimum average response time is when jobs are run by shortest remaining time first algorithm.

Sequence will be R(2), T($3 < x < 5$), S(5), Q(7), P(8).

27. **(b)**

Two binary semaphore Q_1 and Q_2 when $Q_1 = 0$, $Q_2 = 1$, $(010)^n$ will be printed.

When $Q_1 = 1$, $Q_2 = 0$

Process P_1 will execute and then Process P_2 execute and print $(100)^n$.

28. **(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.

29. **(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.

30. **(a)**

In multilevel feedback queue each queue may use different scheduling algorithm and each queue have different priority.

File allocation table does not maintain the list of free block, it is maintained by the super block.

■■■■