

Reg. No. :

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Question Paper Code : 11173

B.E./B.Tech. DEGREE EXAMINATIONS, APRIL/MAY 2026.

Fourth Semester

Computer Science and Engineering

CS3451 — INTRODUCTION TO OPERATING SYSTEMS

(Common to Information Technology)

(Regulations 2021)

Time : Three hours

Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. What are the main objectives of an operating system?
2. Define system call and give two examples.
3. Define the term “process” and list its key characteristics.
4. What is semaphores? Give an example.
5. Define paging. Why is it necessary?
6. Give a scenario where a deadlock might occur in a banking transaction system and suggest a prevention method.
7. What is file system mounting, and why is it necessary?
8. How does the OS manage file sharing and protecting from others?
9. Differentiate between system virtual machines and process virtual machines with suitable example.
10. Compare the mobile OS with the Desktop OS.

PART B — (5 × 13 = 65 marks)

11. (a) Discuss the different operating system structures, including monolithic, layered, microkernel, and modular designs. Compare their advantages and disadvantages.

Or

- (b) Describe the key steps involved in the design and implementation of an operating system. Discuss how structuring methods help in OS development.

12. (a) Suppose that the following jobs arrive as indicated for scheduling and execution on a single CPU.

Job	Arrival Time	Size (msec)	Priority
<i>J1</i>	0	12	1
<i>J2</i>	2	4	3
<i>J3</i>	5	2	1
<i>J4</i>	8	10	3
<i>J5</i>	10	6	2

Draw a Gantt Chart to illustrate how these processes would be scheduled using First Come - First - Serve (FCFS), Shortest Job First (SJF), Shortest Remaining Time First (SRTF), Round Robin (RR - No Pre-emption) and Priority (preemptive) Scheduling. You can assume that the quantum is set to 4 msec. For each scheduling calculate algorithm Average waiting time. Identify the best scheduling algorithm that gives Minimal Waiting Time.

Or

- (b) (i) Explain the difference between a user-level thread and a kernel-level thread? Which scheduling policy is applied to user-level threads? (6)
- (ii) Why in a busy system with several processes in memory, the ready queue becomes empty, and the CPU becomes idle? How does the operating system handle this situation to increase CPU utilization? (7)

13. (a) (i) Consider a system with fixed partition scheme. Given five memory partitions of 100 KB, 500 KB, 200 KB, 300 KB and 600 KB in order, how would the first-fit, best-fit and worst-fit algorithm work with incoming processes of 221KB, 471KB, 127KB, and 451 KB? Which algorithm makes the most efficient use of memory? (7)
- (ii) Consider a memory system with a cache access time of 10 ns and a memory access time of 110ns—assume the memory access time includes the time to check the cache. If the effective access time is 10% greater than the cache access time, what is the hit ratio H? (6)

Or

- (b) Define Copy-On-Write (COW) in the context of virtual memory. Explain with an example how copy-on-write is used during process creation to efficiently manage memory.
14. (a) (i) Describe various file allocation methods used in a file system with each of their strengths and weaknesses. (7)
- (ii) Explain the various techniques adopted for free space management. (6)

Or

- (b) (i) Explain the various disk scheduling algorithms with a neat diagram considering the following set of data: Queue 98, 183, 37, 122, 14, 124, 65, and 67 assuming that the position of the cylinder is at 53. (8)
- (ii) Discuss the relative pros and cons of sector sparing and sector slipping. (5)
15. (a) Explain the various system components of Windows in detail with neat sketch.

Or

- (b) (i) Exemplify the components of the Linux system with neat sketch. (8)
- (ii) How would you classify Linux threads as user-level threads or as kernel-level threads? Support your answer with the appropriate arguments. (5)

PART C — (1 × 15 = 15 marks)

16. (a) Explain the page replacement concept in virtual memory management. Compare two-page replacement algorithms (e.g., FIFO and LRU) using an example. Also, discuss different strategies for allocation of frames to processes.

Or

(b) Consider the following snapshot of a system.

Process	Allocation				Max				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P0	3	0	1	1	4	1	1	1	1	0	2	0
P1	0	1	0	0	0	2	1	2				
P2	1	1	1	0	4	2	1	0				
P3	1	1	0	1	1	1	1	1				
P4	0	0	0	0	2	1	1	0				

- (i) What is the content of the Need matrix?
- (ii) Is the system in a safe state? If yes, give a safe sequence of processes. If not, explain why the system is not in a safe state. (5)
- If a request from process P1 arrives for (0, 0, 1, 0), can the request be granted immediately? Please state the reason. If after this, another request from process P4 arrives for (0, 0, 1, 0), can this request be granted immediately? Please state the reason. (10)