Date / /

Page No.

Name → Aditya Singh

Cource → B.Tech CSE - D

Roll NO → 2301010229

Operating System Assignment 01

PART A

Ques 1 Ans Modern systems still rely heavily on operating system because :-

- Resource mananagement → The OS efficiently manager hardware resources (cpu, memory, storage, I/O devices) and allocates them to different programs.

User & Application Interface → The OS provides a convenient interface b/w hardware and users / applications, enabling portability multitasking and securing.

Q2 Ans Real - Time operating System (RTOS) RTOS ensures timely, predictable and reliable response to inputs like heart rate signals proces management on small, low power hardwares - critical for health monitoring devices

Ques 3 Avoid a monolithic kernel while it give fast system calls, they lack modulaviry and are harder to main - tain / debug. A bug in one service can crash the whole system making them unreliable for critical systems

TOPIC ..... DATE.....

Quest 4
Ans 4 A developer ——— reasoning.
Refute the claim, because OS structure directly impacts performance, reliability, scalability and security.
For ex:- microkernel isolates services for fault tolerance, while a layered structure improves maintainability. Just "running processes" isn't enough if the system is low, insecure or unstable.

Quest 5
i) Explain ——— states.
The PCB stores CPU registers, program counter, state, and memory info. By examining it, we can detect misinitialized registers, wrong state flags, or incorrect program counter values that cause faulty switching.

ii) If a ——— involve?
When a task unexpectedly moves from running to waiting, content switching saves the current process state (registers, program counter, PCB updates) and loads the state of the next process. It ensures execution resumes correctly later.

iii) The ——— and why?
Use an asynchronous, non-blocking system call because this allows the process to continue execution while the I/O is allocated in the background, preventing the CPU from idling.

TOPIC ..... DATE.....

Part B

Ques 6 content ___ performance

a) Total content switching time,
save state = 2ms
load state = 3ms
Scheduler overhead = 1ms

Total time = 2 + 3 + 1 = 6 ms

b) Explain ___ performance.

- content switching is pure overhead (no useful work is done during this time)
- Higher switching time reduces CPU efficiency, as more time is spent switching than executing processes.
- In multitasking, frequent content switches with high overhead can slow down throughput and increase response time.

Ques 7.

- Given:- execution time (single-threaded) = 40 sec
Multithreading is used with n threads per process

Execution time estimate:
In ideal conditions (perfect parallelism, no overhead):

$$T_{multi} = \frac{T_{single}}{n} = \frac{40}{n} \text{ seconds}$$

Example:
if $n = 2 \rightarrow 20$ sec
if $n = 4 \rightarrow 10$ sec
if $n = 8 \rightarrow 5$ sec

TOPIC ..... DATE .....

How multithreading improves performance

- It improves performance by running tasks in parallel, reducing execution time.
- It keeps the CPU busy even during I/O waits, avoiding idle time.
- Threads share resources, making execution faster and more efficient.

Ques Given ____ times

Process : P1 P2 P3 P4

Burst time : 5 3 8 6

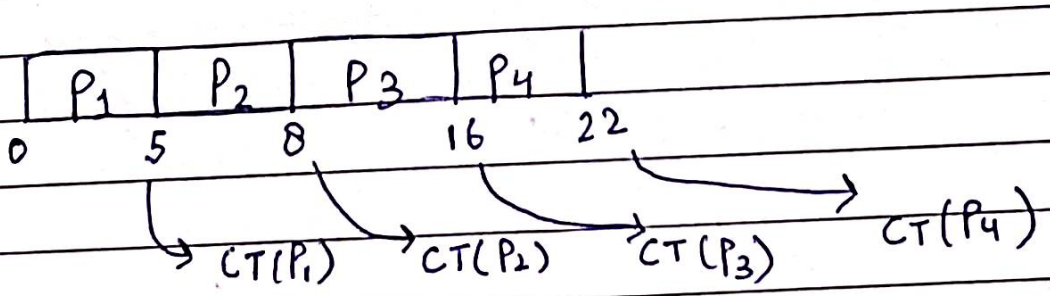
(a) FCFS

Process	Arrival time (AT)	Burst Time (BT)	Completion Time (CT)	Waiting Time (WT)	TAT
P1	0	5	5	5-5=0	5-0=5
P2	0	3	8	8-3=5	8-0=8
P3	0	8	16	16-8=8	16-0=16
P4	0	6	22	22-6=16	22-0=22

WT = Turnaround - Burst (TAT - BT)

TAT = completion - Arrival (CT - AT)

Grantt chart,



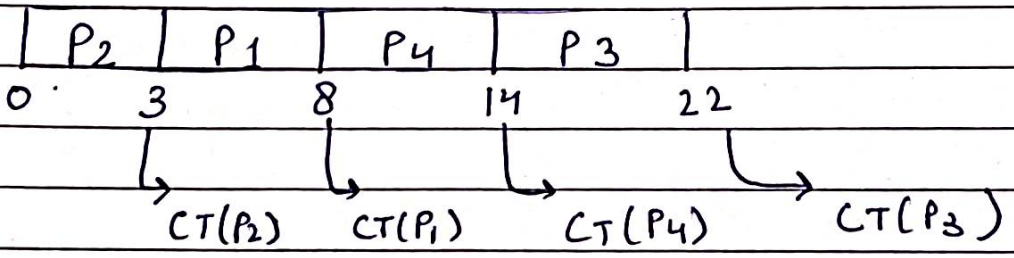
Avg waiting time = (0+5+8+16)/4 = 7.25 ms
Avg turnaround time = (5+8+16+22)/4 = 12.75 ms

(b) Non-preemptive SJF

Process	AT	BT	CT	WT	TAT
P_1	0	5	8	8-5 = 3	8 = 8-0
P_2	0	3	3	3-3 = 0	3 = 3-0
P_3	0	8	22	22-8 = 14	22 = 22-0
P_4	0	6	14	14-6 = 8	14 = 14-0

TAT = CT - AT
WT = TAT - BT

Gyraitt chart,



Avg waiting time = (3+0+14+8)/4 = 6.25 ms
Avg turnaround time = (8+3+22+14)/4 = 11.75 ms

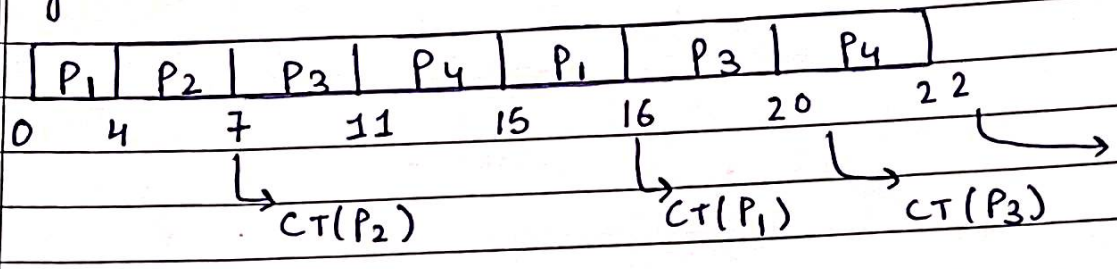
TOPIC .....

(c) Round Robin (quantum = 4ms)

Process	AT	BT	CT	WT	TAT
P_1	0	5	16	16-5 = 11	16-0 = 16
P_2	0	3	7	7-3 = 4	7-0 = 7
P_3	0	8	20	20-8 = 12	20-0 = 20
P_4	0	6	22	22-6 = 16	22-0 = 22

WT = TAT - BT
TAT = CT - AT

Gantt chart,



Avg waiting time = (11+4+12+16)/4 = 10.75 ms
Avg turnaround = (16+7+20+22)/4 = 16.25 ms

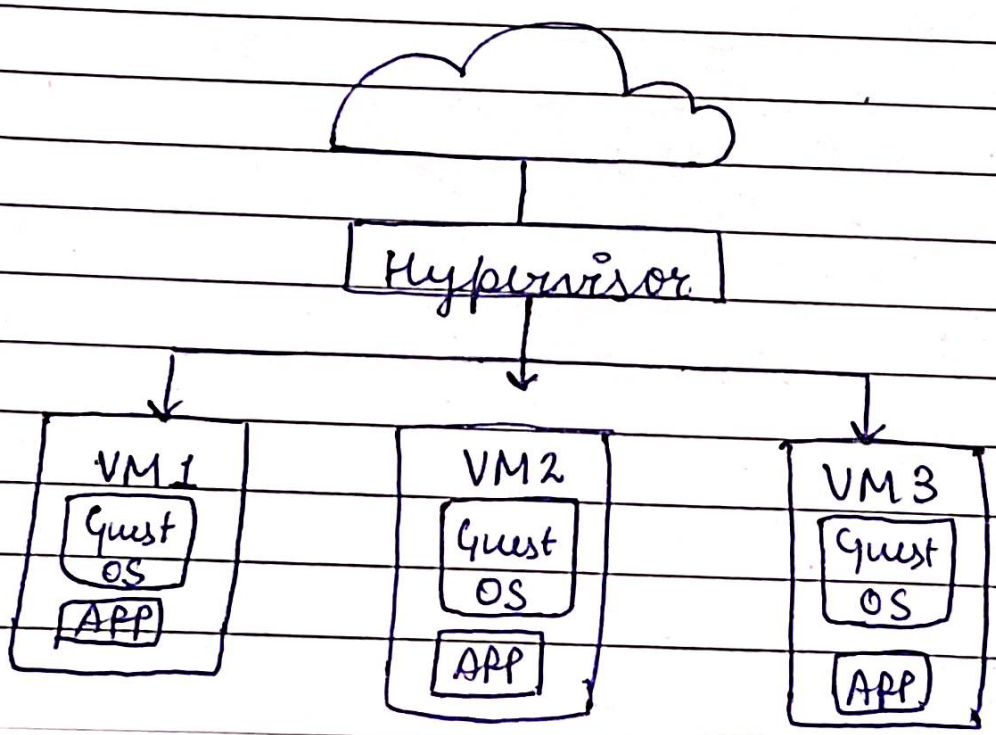
* Non-preemptive SJF is best because it gives the lowest average waiting time (6.25 ms) and turnaround time (11.75ms), while throughput remains the same (4/22) in all methods. It balances turnaround and throughput better than FCFS or RR.

Ques 9. i) Virtualized cloud Migration

(a) Microkernel Architecture would be the best choice because,
High scalability → services (like drivers, file system, networking) run in user space and can be extended / updated easily.
High security & fault isolation → failure in one service doesn't crash the entire system.

(b) Role of Virtual machines in Migration

- Isolation → each VM runs its own OS, preventing one service failure or attack from affecting others.
- Management → VMs can be created, paused, or migrated dynamically across servers for load balancing.
- Resource optimization → Hypervisor allocates CPU, memory, and I/O efficiently among VMs, ensuring better utilization and reducing wastage.



DATE

ii) Smart Home System (IOT devices)

(a) OS role with scheduling + IPC

- Process scheduling → The OS assigns higher priority to critical tasks (eg. intrusion detection) so they preempt less urgent ones (like lighting)
- Inter-Process communication (IPC) → Enables devices and controller processes to exchange data quickly (eg: camera sends motion alert → controller process reacts immediately).

(b) Suitable scheduling Algorithms

- priority scheduling → ensures urgent processes like security alerts preempt routine tasks.
- Earliest Deadline First (EDF) → Useful when tasks must complete before specific deadlines.
- Rate-Monotonic Scheduling (RMS) : Suitable for periodic tasks, like sensor checks.

Among these, PS or EDF are most effective, as they gaurantee timely execution of safety-critical events while maintaining system responsiveness.