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 blw hardware and users / applications, enabling portability multitasking and security.

Q 2 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 modularity and are harder to main - tain / debug. A bug in one service can crash the whole system making them unreliable for critical systems.

Date / /
Page No.

Ques & Ans refute the claim, because OS structure directly impacts performance, reliability, scalability and security.

For ex:- microkernal isolates services for fault tolerance, while a layered structure improves maintainability.

Just "running process" isn't enough if the system is low, unsecure or unstable

Ques

1. The PCB stores cpu register, program counter state and memory ito By enaning it we can detect misinitialized register, corong state flags incorrect program counter values that cause faulty switching.

(ii) When a task unenpectedly moves from running to waiting context switching saves the current process state (register program counter, PCB updates and loads the state of the next process. It ensures execution resumes correctly later.

Date / /
Page No.

(iii) 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

PART B

Ques 1

a) Total content switching time,
Save state = 2 ms
Load State = 3ms
Scheduler overhead = 1ms
Total time = 2+3+1 = 6ms

b) 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 process

In multitasking, frequent content switches with high overhead can slow down throughput and increase response time.

Ques 7 Execution time (single threaded) = 40 sec
Multithreading is used with x threads per process

Execution time estimate :

Date / /
Page No.

In Ideal condition (perfect parallelism no overhead):

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

Example
if n = 2 → 20 sec
if n = 4 → 10 sec
if n = 8 → 5 sec

How multithreading improves performance
- It improves performance by running task in parallel reducing execution time.
- It keeps the cpu busy even during I/O waits avoiding idle time
- Threads shares resources, making execution faster and more efficient.

Ques 8 Process : P_1 P_2 P_3 P_4
Burst time 5 3 8 6

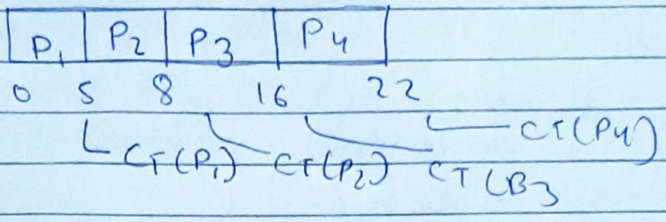
(a) FCFS

Process	Arrival time (AT)	Burst time	completion Time(CT)	waiting Time(WT)	TAT
P_1	0	5	5	5-5=0	5-0=5
P_2	0	3	8	8-3=5	8-0=8
P_3	0	8	16	16-8=8	16-0=16
P_4	0	6	22	22-6=16	22-0=22

Date / /
Page No.

WT = Turnaround-Burst (TAT-BT)
TAT = completion-Arrival (CT-AT)

Grant chart,



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

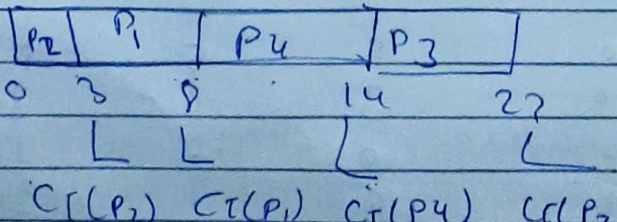
(b) Non-premtive SJF

Process

Process	AT	BT	CT	WT	TAT
P1	0	5	8	8-5=3	8=8-0
P2	0	3	3	3-3=0	3=3-0
P3	0	8	22	22-8=14	22=22-0
P4	0	6	14	14-6=8	14=14-0

TAT = CT-AT
WT = TAT-BT

Grant chart



Date / /
Page No.

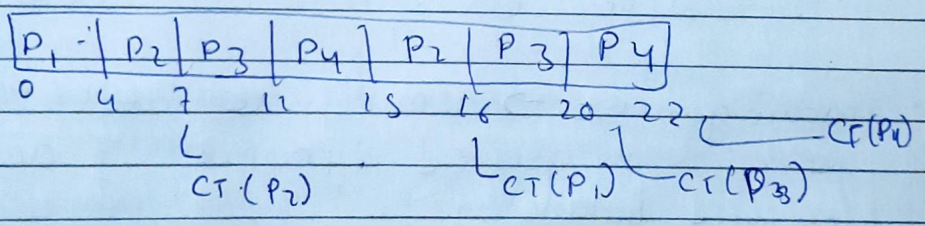
Avg waiting time = (3+0+14+8)/4 = 6.25 ms
Avg turnaround time = (8+3+22+14)/4 = 11.75 ms

(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

Grantt chart



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

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

Date / /
Page No.

Ques i) virtualized cloud migration

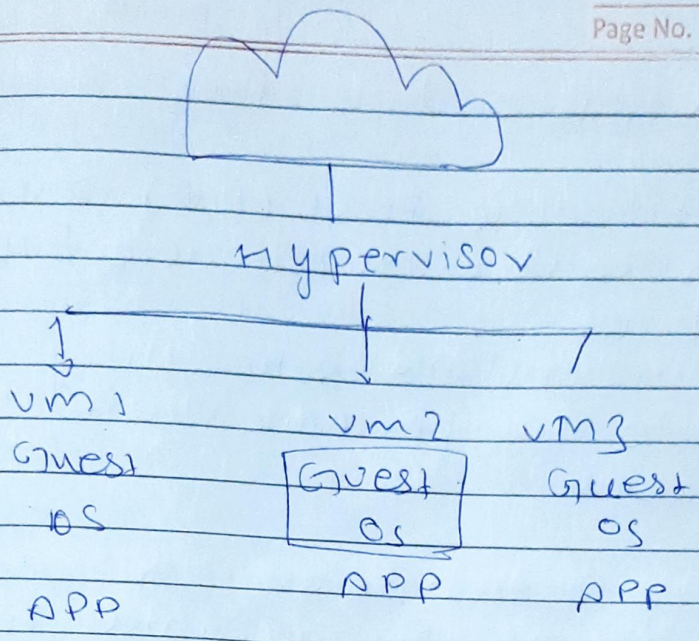
a) Microkernel Architecture would be the best choice because

High scalability → services (like drivers files system, networking, run the 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 run its own OS, preventing one service failure or attack from affect others
- management → VM 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 / /
Page No.



(ii) Smart home system (IoT devices)

(a) OS ~~sched~~ role with scheduling ~~[illegible]~~ + 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 process to exchage data quickly (eg: camera sends motion alert → controller process reacts immediately)

b) suitable scheduling Algoriths

- priority scheduling → ensures urgent process like security alerts prempt

Date / /
Page No.

Aperiodic tasks

Earliest deadline First (EDF) → useful certain tasks must complete before specific deadlines

- Rate-monotonic scheduling (RMS): Suitable for periodic tasks, like sensor checks

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