Date / /
Page No.

Harman Jeet Singh
2301010252
B.Tech CSE - D

Operating System Assignment 01

(Q.1) Despite ---- Systems?

(Ans.1) Modern systems still rely heavily on operating systems because:-

- Resource Management :-

The OS efficiently manages hardware resources (CPU, memory, storage, I/O devices) and allocates them to different programs

- User & Applications Interface :- The OS provides a convenient interface b/w hardware and users/applications, enabling portability multitasking and security

(Q2) You ----- why?

Real time operating System (RTOS)

RTOS ensures timely, predictable and reliable response to inputs like heart rate signals processes data with low latency, provide efficient resource management on small, low-power

Date / /
Page No.

hardware critical for health monitoring devices

(Qs) Given ---- why?

Avoid a monolithic kernal, while it give fast System calls, they lack modularity and are harder to maintain /debug. A bug is one service can crash the whole system, making them unreliable for critical systems

(Or) A developer ---- reasoning

Ans Refute the claim, because as structure directly impacts performance, mediability, scability and security.

for eg:- microkernal isolates services for fault tolerance, while a layred structure improves maintainability, Just "running"

(Qs)

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, incorrect program counter values that cause fault switching

Date / /
Page No.

(ii) (a) A of a ---- involves

When a task unexpectedly moves from running to waiting, context switching saves saves the current process state (registers, program counter, PCB updates) and loads the state of the next process.

(b) Context ---- ~~prog~~ performance

a) Total context switching time,
Save state = 2ms
Load state = 3ms
Schedulen overhead = 1ms

Total time = 2+3+1 = 6ms

b) Explain ---- performance

- context switching is pure overhead (no useful work is done durning this time)
- Higher switching time reduces CPU efficiency as more time is spent switching then executing process
- In multi tasking, frequent conduct switches with high overhead can slow down throughput and increase response time

Date / /
Page No.

(Q7) execution time (single - threaded) = 40 sec
multi threading is used with n threads per process

Execution time estimate:
In idol condition (perfect parall time, no overhead):

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

Example:

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

How multi threading improves performance

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

(or) Given - - - - times

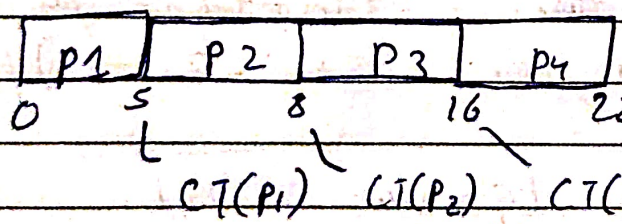
Process :	P1	P2	P3	P4
Burst time :	5	3	8	6

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)

Gantt chart



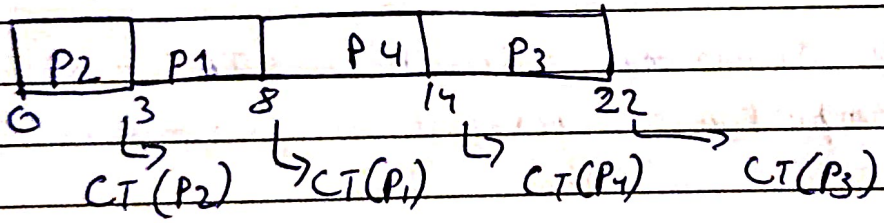
Avg waiting time = (0+5+8+16)/4 = 7.25ms
Avg turnaround time = (5+8+16+22)/4 = 12.7ms

(b) Non-preemptive SJF

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

Grantt chart,



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

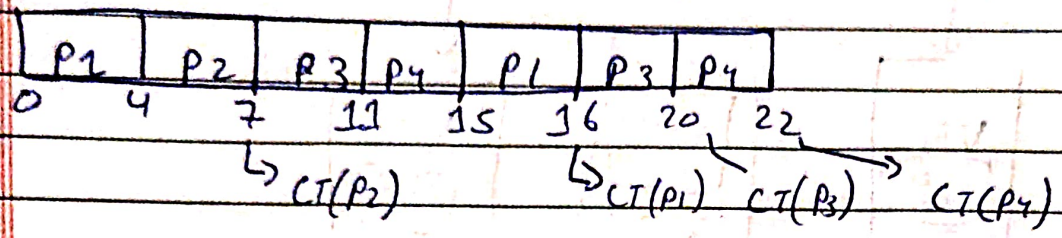
(c) Round Robin (quantum = 4ms)

Process	AT	BT	CT	WT	TAT
P1	0	5	16	16-5=11	16-0=16
P2	0	3	7	7-3=4	7-0=7
P3	0	8	20	20-8=12	20-0=20
P4	0	6	22	22-16=16	22-0=22

Date / /
Page No.

$WT = TAT - BT$
$TAT = CT - AT$

Gantt chart,

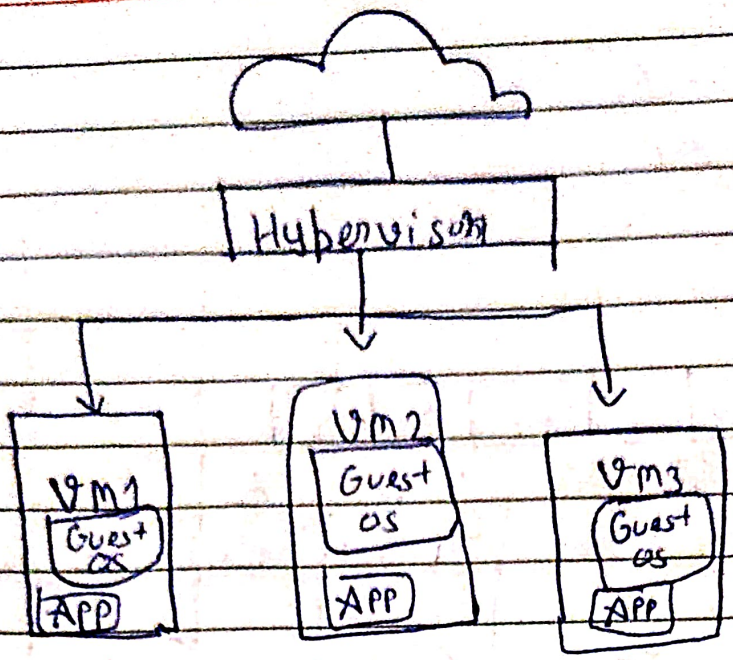


Avg waiting time = $(11 + 4 + 12 + 16)/4 = 10.75$ ms
Avg turnaround = $(16 + 7 + 20 + 22)/4 = 16.25$ ms

(Q9) i) Virtualized cloud Migration

(a) Microkernal Architecture would be the best choice because,

High scalability → services (like driver, file systems, 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.



(ii) Smart Home System (IoT devices)

(a) OS role with scholduting + IPC

- Process scheduling ⇒ The OS assigns higher prioriity to critical tasks (eg. intrusion detection) so they prempt less urgent ones (like lighing)
- Inter-process communcication (IPC) → Enables devices and controller processes to exchange data quickly (eg: camera sends motion alert → controller process reacts immediatly)