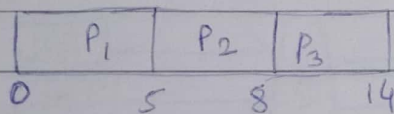***** **DHCP:-** Dynamic host configuration protocol

Q. Consider the following processes with arrival times and burst times: calculate avg waiting period for FCFS.

Process	Arrival Time	Burst Time	Waiting Period = (CPU Allocation – Arrival time
P_1	0	5	0
P_2	1	3	4
P_3	2	6	6.

Gantt chart

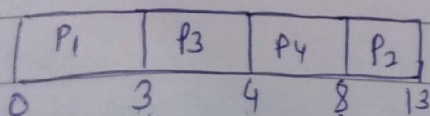| P_1	P_2	P_3
0 5	8	14

∴ avg waiting period

$$= \frac{\text{sum of waiting period}}{\text{no. of processes}}$$

$$= \frac{0+4+6}{3} = \frac{10}{3} = 3.3$$

Q.2 Consider the following processes with arrival times & burst times. calculate the average turnaround time using SJF.

Process	Arrival Time	Burst Time.	completion Time	Waiting Time	Turnaround Time
P_1	0	3	3	0	3
P_2	1	5	13	7	12
P_3	2	1	4	1	2
P_4	3	4	8	1	5

Gantt chart

P_1	P_3	P_4	P_2
0 3	4	8	13

Avg waiting time = $8/4 = 2$

Avg Turnaround time = $22/4 = 5.5$

Scanned with CamScanner

Q.3

Consider the following processes with arrival times & burst times and priorities, calculate avg waiting time using priority sched.

Process	Arrival Time	Burst Time	Priority	waiting time
P_1	0	6	3	5
P_2	1	4	1	0
P_3	2	7	4	9
P_4	3	2	2	1

Gantt chart

P_1	P_2	P_4	P_1	P_3

0   1   4   6   11   18
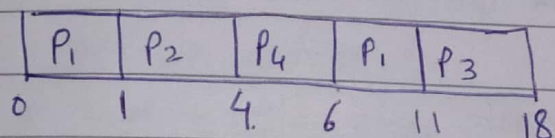
$$\text{Avg waiting Time} = \frac{5+0+9+1}{4}$$

$$= \frac{15}{4}$$

$$= \boxed{3.7}$$

Q.4

Consider the following process with arrival times & burst times, and the time quantum for Round Robin scheduling is 2 units. Calculate avg turnaround time.

Process	Arrival Time	Burst Time	Completion Time	Turnaround Time
P_1	0	4	10	10
P_2	1	5	14	13
P_3	2	2	6	4
P_4	3	3	13	10

Gantt chart

P_1	P_2	P_3	P_4	P_1	P_2	P_4	P_2

0   2   4   6   8   10   12   13   14
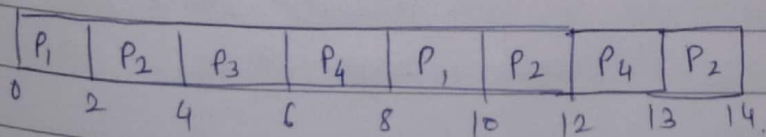
$$\text{Avg Turn Around time} = \frac{10+13+4+10}{4}$$

$$= \frac{37}{4} = \boxed{9.25}$$

Scanned with CamScanner

Q. what is os & its primary functions —

→
- It is interface bet^n user and hardware.
- OS is a software that manages the software hardware.
- It is a hardware resource manager.
- It is process manager.
- It takes the input from the user & instruct the hardware to produce desired o/p.

* functions of os :-

- Memory management. -
- Process management
- File management.
- Network management
- Security management
- Disk management.

Q. Diff. bet^h process & Thread

Process	Thread
- Process means any prog. is in execution.	- Thread is a segment of process
- Process takes more time for creation and termination.	- Thread takes less time for creation & termination compared to thre
- The process is isolated.	- Threads share memory.
- System call involved in it.	- No system call involved, it is created using API's.
- process does not share data with each other	- Threads share data ~~which~~ with each other.

Q.3 Virtual memory & how it works?

- It is memory in hard-drive, which works like physical memory (RAM) to entertain large processes which having size bigger than the RAM. It is an illusion RAM.

- In virtual memory process is divided into fixed size partitions known as pages.

- Os loads the processe's page from virtual memory to physical memory on demand of CPU.

- while working with virtual memory pages are saved as per their logical address.

- while loading the pages from virtual memory to physical memory pages gets the physical address.

- It page is loaded from virtual memory to physical memory known as swap-in process.

- when page is shifted from physical to virtual memory by the replacement process known as swap out.

Q.4 Diff in multiprogramming, multitasking & multiprocessing.

i) Multiprogramming :- Running multiple program on a single CPU. works on context switching.
 - Uses Priority & Round Robin scheduling Algo.

ii) Multitasking :- Running multiple tasks at a same time.
 eg we can able to play mp3 songs, edit docs at same time.
 - It is logical extension of multiprogramming

- Multiprocessing -

- having more than one CPU. for performance improving.
- It achieves max. throughput.
- used for load balancing.

Q.5 what is file system & its components
→ - File system is a method of os used to store, organize & manage files and directories stored on storage device.

Q.6 Deadlock & its prevention :-

→ - It is a condition ~~when~~ where os can't continue its executions because process demanding the resources which are hold by other process.
- So none of the process can continue its execution. hence the deadlock occurs.

- It can be prevented by →
  1. No mutual exclusion.
  2. using pre-empitive System

Q.7 Diff in kernal & shell.

→

shell - It is special program which provides interface bet^w the user & kernal.
  - It executes the program based on the input provided by the user.

- **kernal :-** Its heart of os. that manages the all the operations of computer & hardware.
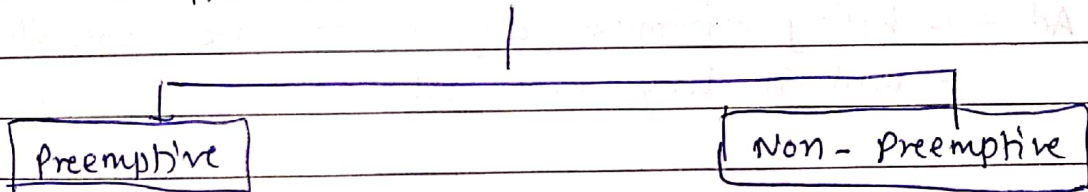
- It a bridge beth user & resources of the system by accessing various computer resources. like cpu, I/o devices etc.

9.8 **What is cpu scheduling why its imp ?**
→
- It is process which allowes one process to use CPU while another process is delayed due to unavailability of any resources such as I/o etc.
- This is imp because CPU can handle only one task at a time, but there are usually many tasks that need to be processed.

* Diff. types of CPU scheduling Algo ⟶

```
                    |
          ┌─────────┴──────────┐
    ┌─────┴─────┐          ┌────┴──────────┐
    │ Preemptive│          │ Non-Preemptive│
    └───────────┘          └───────────────┘
```

o) **Preemptive** → used when process goes from running state to ready state or from waiting to ready state.

•) **Non-Preemptive :-** used when whole processes is completed and terminated (stop) or goes from running to writing state

Algos →

i) FCFS → first come first serve
   - Supports ~~pre em~~ preemptive & non-premptive
   - easy to implement.
   - not much efficient in performance.
   - high waiting time.

   Dis → FCFS suffers from Convoy effect.

     Phenomenon ↗ where a few slower processes slow downs
     the entire system.

2. SJF → shortest job first.
   - selects the waiting process with smallest burst time

   Adv → - having minimum avg. waiting time from all.
         - used for long term scheduling.

   Dis Adv → - If shortest process keep coming leads to starvation

3. Priority scheduling — schedule task on basis of priority.

4. Round Robin → A fixed slice of time is given to
   each & every process. The slice of ~~size~~ time known
   as quantum.

Q. Thrashing & how to avoid it –

→ – Thrashing is a condition when the system is spending a major portion of its time servicing the page fault, but the actual processing done is very negligible.

Q semaphore & its use in synchronization –

→

– Semaphores are just normal variables used to co-ordinate the activities of multiple processes in a computer system.

– They are used to enforce mutual exclusion, avoid race conditions, and implement synchronisation between processes,

– It uses signaling for synchronisation.

– It provides two operation → wait (w) & signal (v)

– The wait decrement the value of semaphore

– The signal increment ——–

. When semaphore equal to zero, any process that performs a wait operation will be blocked until another process performs a signal operation