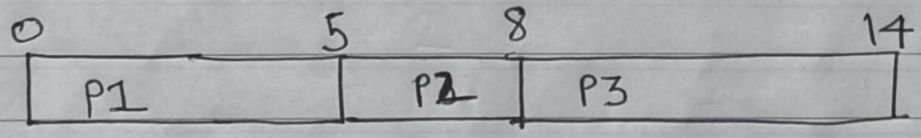
Name: Shriram Sabade
Module: Concepts of OS

Assignment – 2 Part – E

Q1] Algorithm used: FCFS

Process	Arrival Time	Burst time	Waiting Time
P1	0	5	0
P2	1	3	4
P3	2	6	6

Gantt chart:

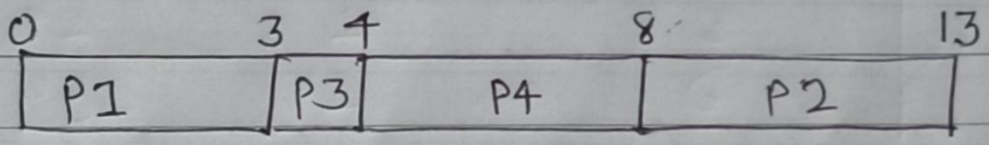


Avg. Waiting Time = (0+4+6)/3
= 10/3
= 3.333333
≈ 3.33

Q2] Algorithm Used : SJF (Non-Preemptive)

Process	Arrival Time	Burst Time	Waiting Time	Turnaround Time
P1	0	3	0	3
P2	1	5	7	12
P3	2	1	1	2
P4	3	4	1	5

Gantt chart :



Avg. Turnaround time = $\frac{3+12+2+5}{4}$

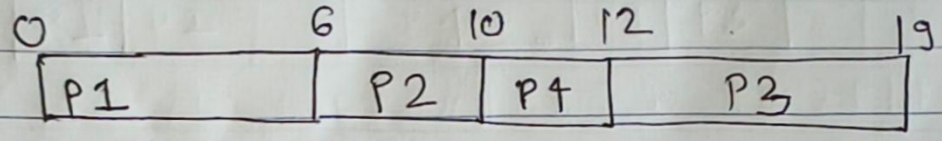
$= \frac{22}{4}$.

$= \underline{\underline{5.5}}$

Q3] Algorithm Used: Priority Scheduling (Non-preemptive)

Process	Arrival Time	Burst time	Priority	Waiting Time
P1	0	6	3	0
P2	1	4	1	5
P3	2	7	4	10
P4	3	2	2	7

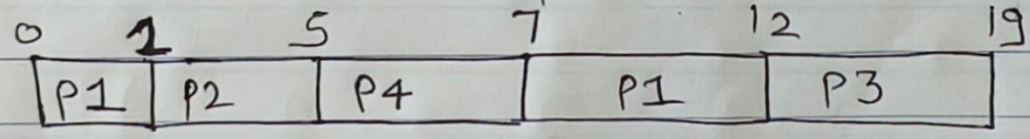
Gantt Chart:



Avg. Waiting Time = $\frac{22}{4}$

$= \underline{5.5}$

Gantt chart (Preemptive):



Waiting time
6
0
2
10

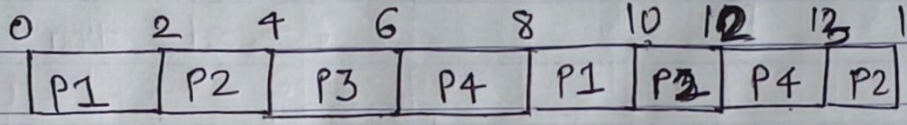
Avg. Waiting time = $\frac{18}{4}$

$= \underline{4.5}$

Q4] Algorithm Used : Round Robin
Quantum = 2 units

Process	Arrival Time	Burst Time	Waiting Time	Turnaround Time
P1	0	4	6	10
P2	1	5	8	13
P3	2	2	2	4
P4	3	3	7	10

Gantt chart: (CPU ~~Idle~~ not kept idle)



Avg. Turnaround Time = (10+13+4+10)/4

= 37/4

= 9.25

- **Q5.**
  - When the fork() system call is used, it creates a child process that has its own copy of the parent's memory.
  - Before forking, the parent has a variable x = 5. After the fork, both the parent and child have separate copies of x, still equal to 5.
  - Each process then increments x by 1, so both the parent and child have x = 6, but in their own separate memory.
  - In parent process, x=6. In child process, x=6