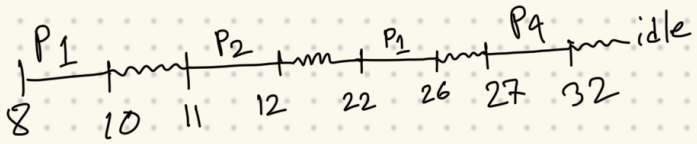
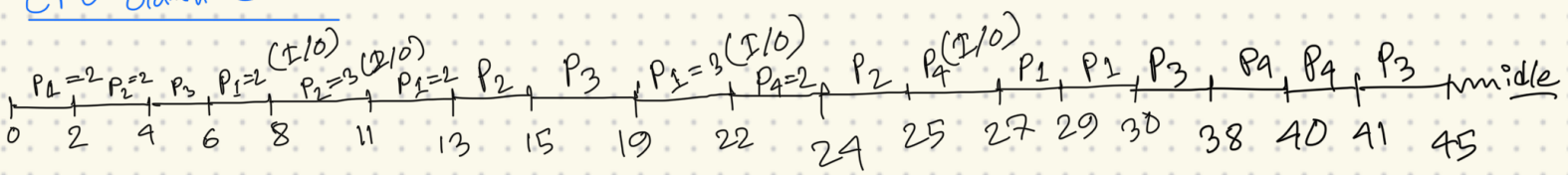
MLFQ Scheduling (I/O)

Processes	Burst Time	Arrival Time	I/O Time
P1	12	0	6 [2s I/O operation after total 4s of CPU allocation & 4s I/O operation after total 9s of CPU allocation]
P2	8	1	1 [1s I/O operation after total 5s of CPU allocation]
P3	18	4	N/A
P4	7	20	5 [5s I/O operation after total 4s of CPU allocation]

I/O Gantt Chart:

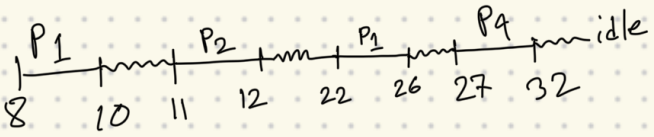


CPU Gantt Chart:

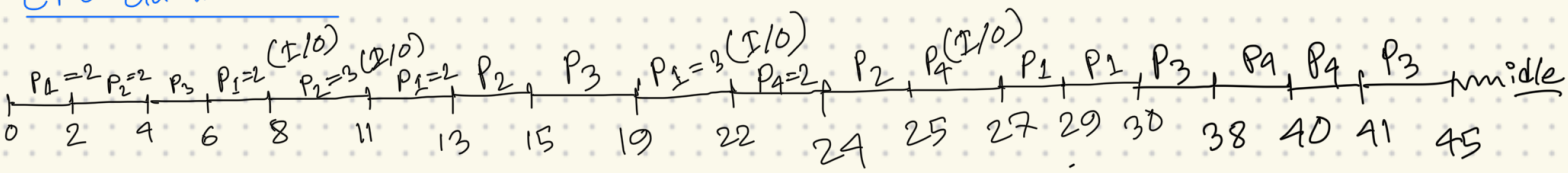


T	(t=2) Q0	(t=4) Q1	(t=8) Q2	I/O
0	P1(12)	X	X	X
2	P2(8)	P1(10)	X	X
4	P3(18)	P1(10), P2(6)	X	X
6	X	P1(10), P2(6), P3(16)	X	X
8	X	P2(6) P3(16)	X	P1(2)8
11	P1(8)	P3(16)	X	P2(1)3
13	P2(3)	P3(16) P1(6)	X	X
15	X	P3(16) P1(6) P2(1)	X	X
19	X	P1(6) P2(1)	P3(12)	
22	P4(7)	P2(1)	P3(12)	P1(4)3
24	X	P2(1), P4(5)	P3(12)	X
25	X	P4(5)	P3(12)	X
27	P1(3)	X	P3(12)	P4(5)3
29	X	P1(1)	P3(12)	X
30	X	X	P3(12)	X
38	P4(3)	X	P3(4)	X
40	X	P4(1)	P3(4)	X
41	X	X	P3(4)	X
45	e	e	e	e

I/O Gantt Chart:



CPU Gantt Chart:



waiting time:

$P_1 = (0-0) + (6-4) + (11-10) + (19-13) + (27-26) + (29-29)$

$= \square$?

Process	Arrival Time	Burst Time	I/O Request
P1	0	10	3s IO after 7s CPU
P2	2	23	4s I/O after 5s CPU 3s I/O after 12s CPU
P3	5	18	N/A
P4	7	16	2s I/O after 11s CPU

3. Given the arrival time and burst time for the processes below, draw a gantt chart using multilevel feedback queue scheduling algorithm showing the states of the ready queues of different levels. Then calculate the average waiting time, response time and turnaround time. There are 3 queues with decreasing priority all using round robin scheduling algorithm internally. **Q1 (quantum: 3, priority:0), Q2 (quantum:5, priority: 1), Q3 (quantum:10, priority:2)**. In case of I/O request, a process will be promoted one level upon I/O completion.

Answer:
Avg. Waiting: 27.25
Avg. Response: 1
Avg. Turnaround: 47.25