Lab Class 3: Scheduling

Consider the following set of processes with arrival times and length of CPU burst times given in milliseconds:

Process	P1	P2	P3	P4	P5
Arrival time	9	8	7	6	5
Burst time	3	9	4	7	9

1. Which CPU scheduling algorithm listed below yields the shortest average turnaround time?

   (a) FCFS (First-Come-First-Served)
   (b) SJF (Shortest Job First; without preemption)
   (c) SRT (Shortest Remaining Time; same as SJF with preemption)
   (d) RR-1 (Round-Robin with a time quantum of 1 millisecond)

2. Which CPU scheduling algorithm listed below yields the shortest average waiting time?

   (e) RR-1 (Round-Robin with a time quantum of 1 millisecond)
   (f) RR-2 (Round-Robin with a time quantum of 2 milliseconds)

Let's reorder the processes according to arrival time first:

Process	P5	P4	P3	P2	P1
Arrival Time	5	6	7	8	9
Burst Time	9	7	4	9	3

---

1. (a) FCFS: only the arrival times matter, so the order will be P5, P4, P3, P2, P1. Therefore the execution intervals will be:

P5: 5 + 9 → **14**, P4: 14 + 7 → **21**, P3: 21 + 4 → **25**, P2: 25 + 9 → **34**, P1: 34 + 3 → **37**

and the turnaround times:

P5: **14** – 5 = 9, P4: **21** – 6 = 15, P3: **25** – 7 = 18, P2: **34** – 8 = 26, P1: **37** – 9 = 28

Therefore the average turnaround time: (9 + 15 + 18 + 26 + 28) / 5 = 96 / 5 = **19.2**

(b) SJF: P5 arrives first so it starts right away and is not interrupted (no preemption):

P5: 5 + 9 → **14**

By the time it is finished (14ms), the four other processes have arrived and were queued in the "Ready" state: they are now going to be unqueued according to burst time, thus: P1, P3, P4, P2. So we get:

P5: 5 + 9 → **14**, P1: 14 + 3 → **17**, P3: 17 + 4 → **21**, P4: 21 + 7 → **28**, P2: 28 + 9 → **37**

and the turnaround times:

P5: **14** – 5 = 9, P1: **17** – 9 = 8, P3: **21** – 7 = 14, P4: **28** – 6 = 22, P2: **37** – 8 = 29

Therefore the average turnaround time: (9 + 8 + 14 + 22 + 29) / 5 = 82 / 5 = **16.4**

(c) SRT: With preemption, we must reconsider scheduling at each arrival time 5, 6, 7, 8, 9. Let's show a snapshot of the Ready queue and Running process at each arrival time step. P1(x) means x is the remaining time of P1, i.e., P1 has x milliseconds more to run.

time 5:	P5(9) arrives	Ready: P5(9)	Running: --	
	⇒P5(9) scheduled	Ready: --	Running: P5(9)	
time 6:	P4(7) arrives	Ready: P4(7)	Running: P5(8)	
	⇒P4(7) scheduled	Ready: P5(8)	Running: P4(7)	
time 7:	P3(4) arrives	Ready: P5(8) P3(4)	Running: P4(6)	
	⇒P3(4) scheduled	Ready: P5(8) P4(6)	Running: P3(4)	
time 8:	P2(9) arrives	Ready: P5(8) P4(6) P2(9)	Running: P3(3)	
time 9:	P1(3) arrives	Ready: P5(8) P4(6) P2(9) P1(3)	Running: P3(2)	
time 11:		Ready: P5(8) P4(6) P2(9) P1(3)	Running: --	P3 exits at **11**

No more processes arrive at this point, so P1(3) gets scheduled next and will not be interrupted since it will always have a shorter remaining time than the other queued processes. Thus, P1 exits at time 11 + 3 = **14**. Then it is the turn of P4(6), which also finishes without interruption at time 14 + 6 = **20**. Then, P5(8) finishes at time 20 + 8 = **28** and P2(9) at time 28 + 9 = **37**.

Thus, the turnaround times are:

P3: **11** – 7 = 4, P1: **14** – 9 = 5, P4: **20** – 6 = 14, P5: **28** – 5 = 23, P2: **37** – 8 = 29

Therefore the average: (4 + 5 + 14 + 23 + 29) / 5 = 75 / 5 = **15.0**

(d) RR-1: Here, we simply follow the order of arrival of the processes and then loop over the processes still present at every time step. When a process has exhausted its burst time, it's simply removed from the merry-go-round. So here it goes (indexes are cycle numbers; here, a line represents a temporal sequence of running processes, *not* a queue).

1. P5 P4 P3 P2 P1 (arrival order: it happened that all processes arrived within 1 of each other)
2. P5 P4 P3 P2 P1
3. P5 P4 P3 P2 P1 (P1 has done 3 rounds and is finished at time **20** = 5 + counting all P's so far)
4. P5 P4 P3 P2 (P3 has done 4 rounds and is finished at time **23** = 5 + counting all P's so far)
5. P5 P4 P2
6. P5 P4 P2
7. P5 P4 P2 (P4 has done 7 rounds and is finished at time **32** = 5 + counting all P's so far)
8. P5 P2
9. P5 P2 (P5 has done 9 rounds and is finished at time **36** = 5 + counting all P's so far)
   (P2 has done 9 rounds and is finished at time **37** = 5 + counting all P's so far)

Thus, the turnaround times are:

P1: **20** – 9 = 11, P3: **23** – 7 = 16, P4: **32** – 6 = 26, P5: **36** – 5 = 31, P2: **37** – 8 = 29

Therefore the average: (11 + 16 + 26 + 31 + 29) / 5 = 113 / 5 = **22.6**

Conclusion:

(a)	FCFS:	**19.2**	
(b)	SJF:	**16.4**	
(c)	SRT:	**15.0**	← SRT yields the shortest average turnaround time
(d)	RR-1:	**22.6**	

---

2. (e) RR-1: the waiting time is the total time spent outside of execution between the moment the process started until it finished. In other words, the waiting time is the turnaround time minus the burst time. Therefore, we get the following waiting times from (d):

P1: 11 – 3 = 8, P3: 16 – 4 = 12, P4: 26 – 7 = 19, P5: 31 – 9 = 22, P2: 29 – 9 = 20

and the average waiting time: (8 + 12 + 19 + 22 + 20) / 5 = 81 / 5 = **16.2**

(f) RR-2: same processus of merry-go-round than (d), except every 2ms instead of 1ms:

1. P5(2) P4(2) P3(2) P2(2) P1(2) (arrival order)
2. P5(2) P4(2) P3(2) P2(2) P1(1) (P3 has executed 4ms and finishes at time 5 + 16 = **21**)
   (P1 has executed 3ms and finishes at time 5 + 19 = **24**)
3. P5(2) P4(2) P2(2)
4. P5(2) P4(1) P2(2) (P4 has executed 7ms and finishes at time 5 + 28 = **33**)
5. P5(1) P2(1) (P5 has executed 9ms and finishes at time 5 + 31 = **36**)
   (P2 has executed 9ms and finishes at time 5 + 32 = **37**)

Thus, the turnaround times are:

P3: **21** – 7 = 14, P1: **24** – 9 = 15, P4: **33** – 6 = 27, P5: **36** – 5 = 31, P2: **37** – 8 = 29

and the waiting times:

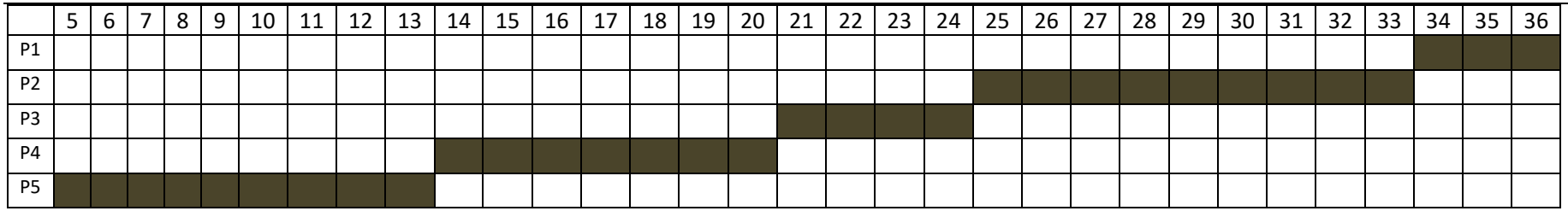
P3: 14 – 4 = 10, P1: 15 – 3 = 12, P4: 27 – 7 = 20, P5: 31 – 9 = 22, P2: 29 – 9 = 20

and the average waiting time: (10 + 12 + 20 + 22 + 20) / 5 = 84 / 5 = **16.8**

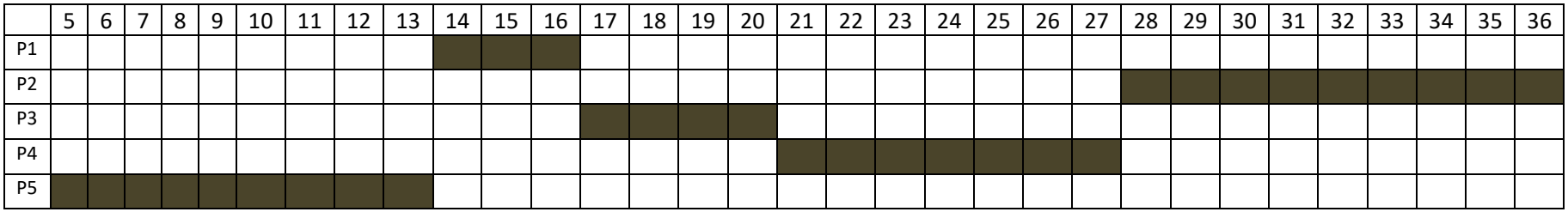
Conclusion:

(e) RR-1: **16.2** ← RR-1 yields the shortest average waiting time
(f) RR-2: **16.8**

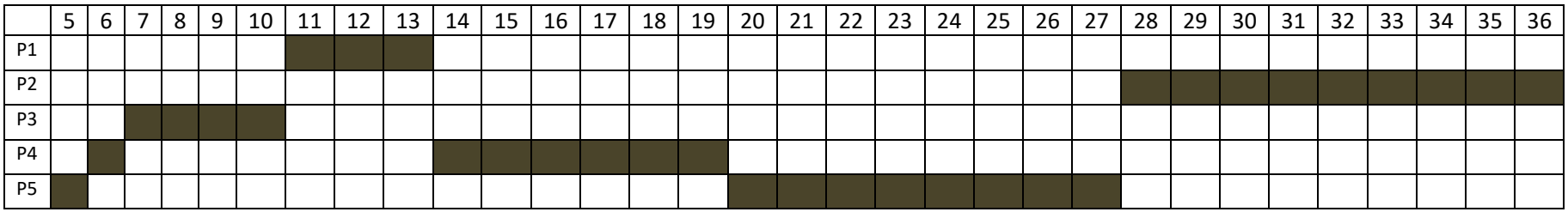
Process	P5	P4	P3	P2	P1
Arrival Time	5	6	7	8	9
Burst Time	9	7	4	9	3



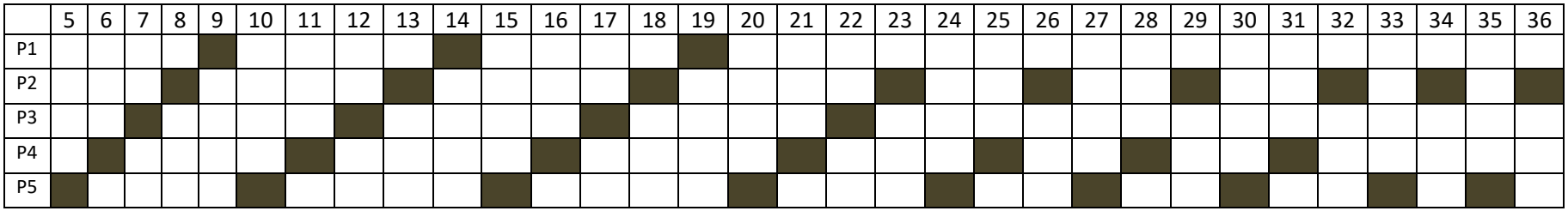
FCFS



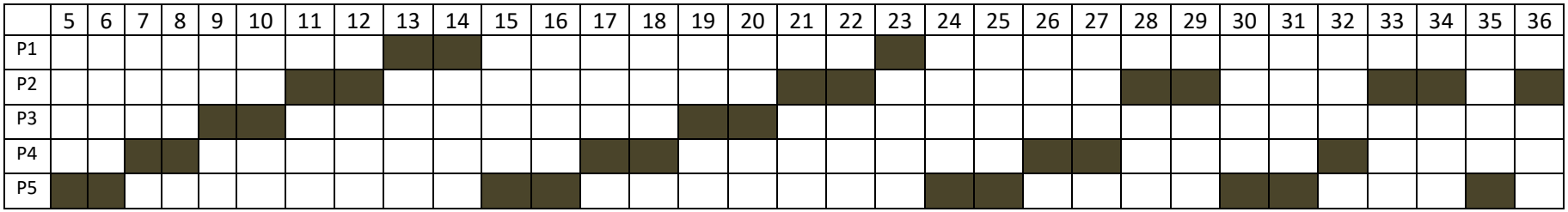
SJF



SRT


5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
P1
P2
P3
P4
P5

RR1


5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
P1
P2
P3
P4
P5

RR2