INGENIERÍA INFORMÁTICA
OPERATING SYSTEMS

Laboratorio Introducción a Procesos

Goals

- Practice with process scheduling concepts

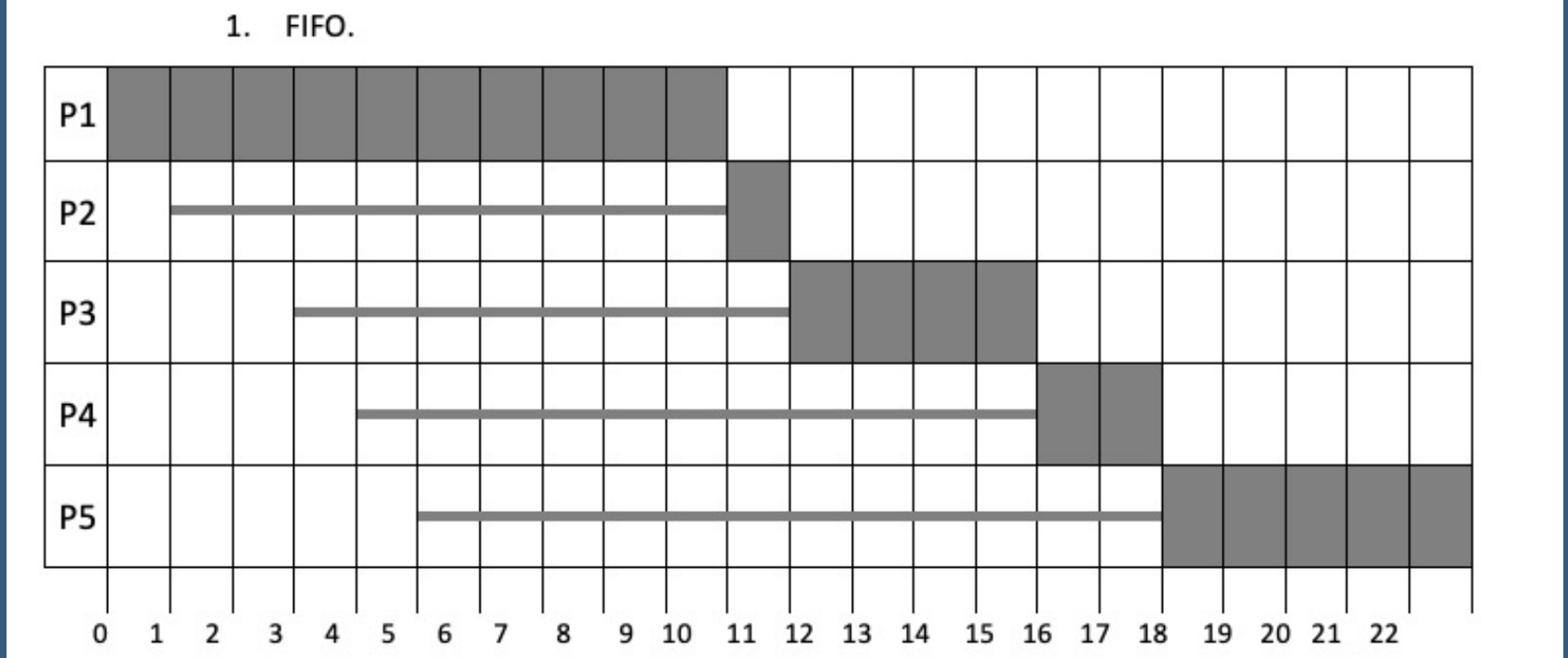
1. Statement

- Given the next set of processes:

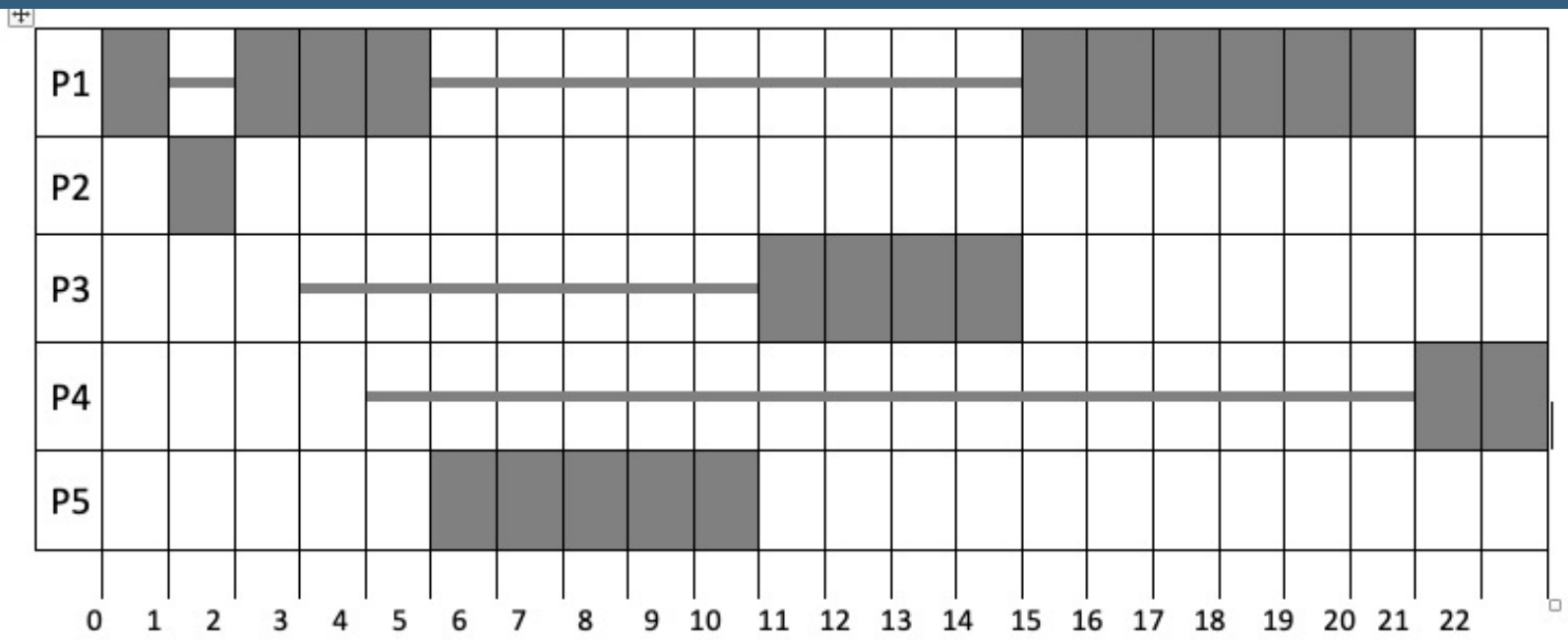
Process	Arrival	CPU Time	Priority
P1	0	10	3
P2	1	1	1
P3	3	4	3
P4	4	2	4
P5	5	5	2

- a) Write a diagram that illustrates the execution of these processes using:
  - 1. FIFO.
  - 2. Scheduling with preemptive (or expulsive) priorities
  - 3. Scheduling with preemptive priorities and with Round Robin (q = 2) for the processes of the same priority.
- b) Calculate the waiting time for each scheduling process and algorithm

1. Solution. FIFO

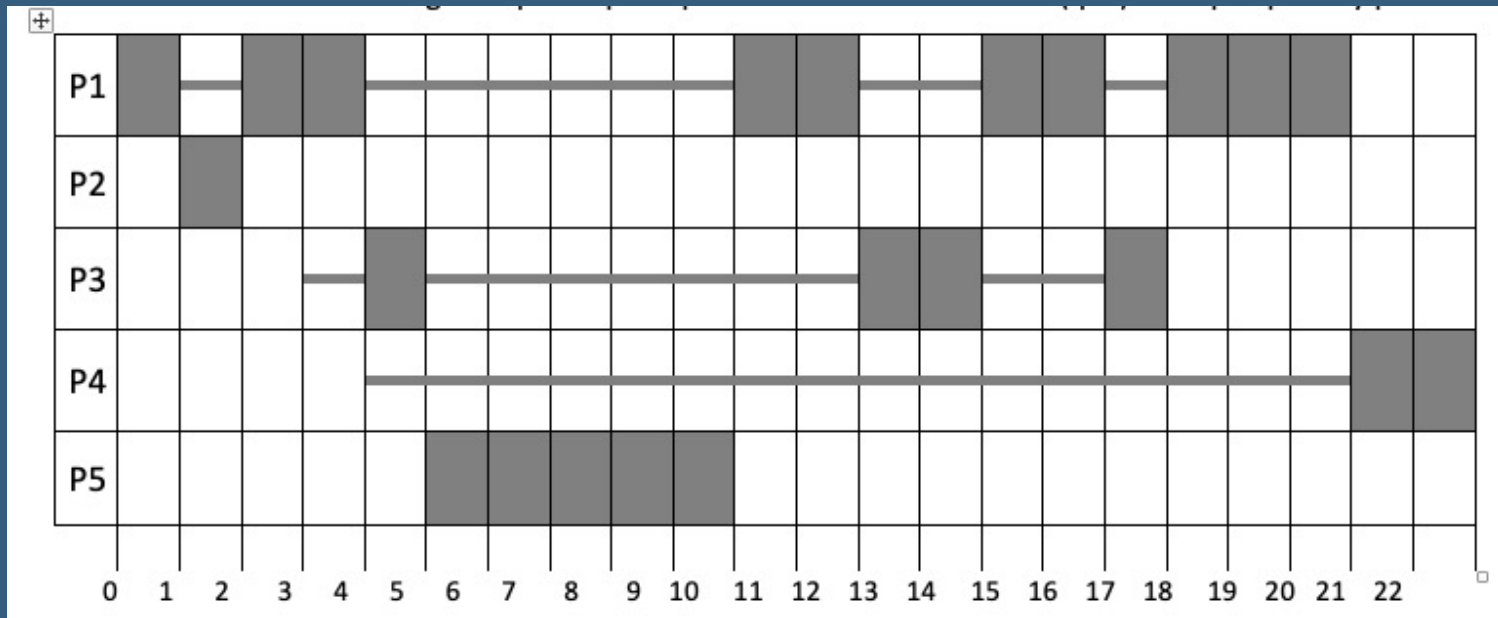


1. Solution. Preemptive



- En el instante t=10, se pone a ejecutar P3 porque estaba a la cola de listos antes que P1.

1. Solution. Preemptive and RR(q=2)



- In time t=10, P1 is run as it was in the ready queue before P3.

1. Solution. Waiting time

	P1	P2	P3	P4	P5
Fifo	0	9	8	11	12
Priorities	10	0	7	16	0
Priorities and Round Robin	10	0	10	16	0

2. Statement

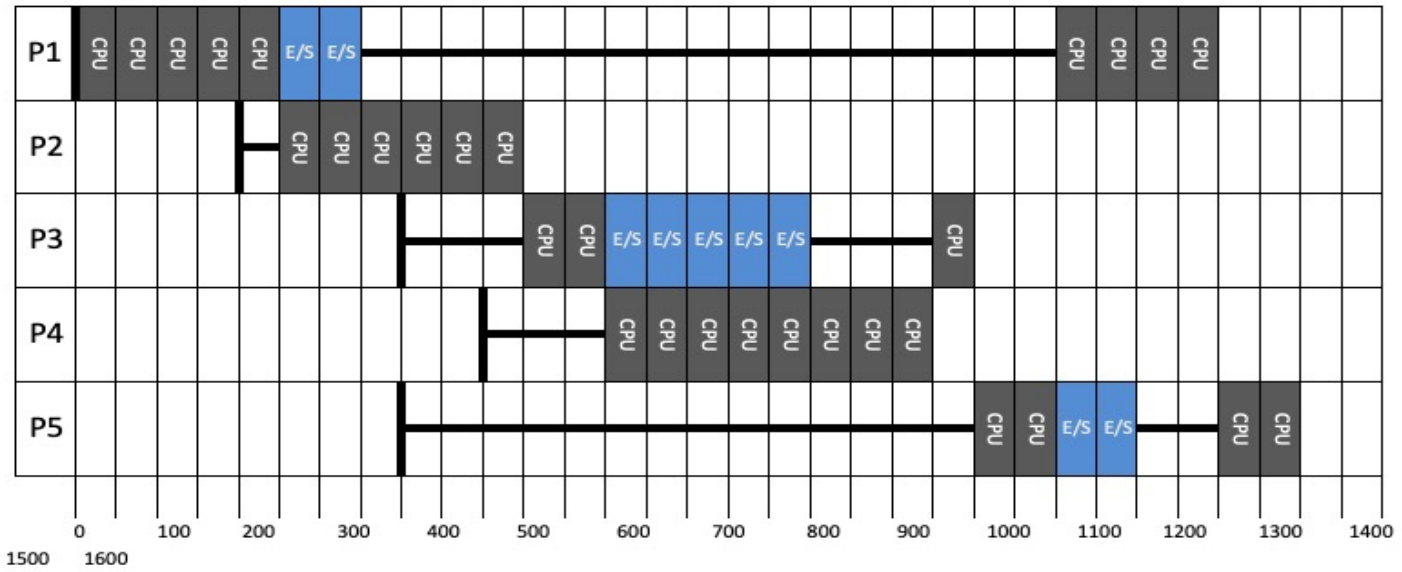
- Scheduling using priority queues (1 being the highest).
- When several processes have the same priority, a round robin scheduling policy is used, with a 100 ms slice.
- For the next table of processes:

PROCESSES	PRIORITY	ARRIVAL	EXECUTION TIME
P1	3	0	250 CPU + 100 E/S + 200 CPU
P2	2	200	300 CPU
P3	1	400	100 CPU + 250 E/S + 50 CPU
P4	1	500	400 CPU
P5	2	400	100 CPU + 100 E/S + 100 CPU

- For scheduling without and with preemption:
  - 1. Make a cronogram of the execution of the processes.
  - 2. Calculate the time that each process is kept on hold from its arrival in the system until it ends.
  - 3. Calculate the return time of each process (time elapsed since the process arrives until the end of its execution).
  - 4. Average waiting time and average return time.

2. Solution. Non preemptive

a) **Non-preemptive scheduling:**

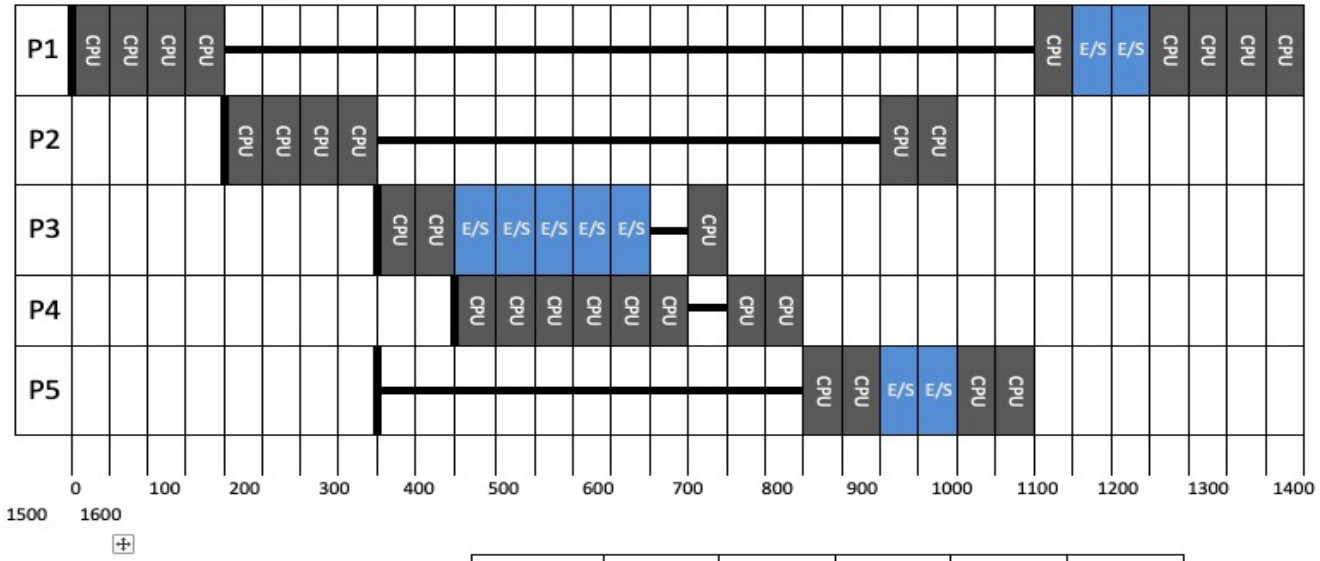


	P1	P2	P3	P4	P5	Avg
Wait time (ms)	850	50	300	150	800	430
Turnaround time (ms)	1400	350	700	550	1100	820

2. Solution. Preemptive

b) Preemptive scheduling:

- The new processes enters in the Ready queue before Preempted processes → example in *950 ms P5 goes before P2*.



	P1	P2	P3	P4	P5	Avg
Wait time (ms)	1050	650	50	50	550	470
Turnaround time(ms)	1600	950	450	450	850	860

3. Statement proposed

- An operating system uses a cyclic (round-robin) scheduler. At a given moment there are no jobs running and you want to run jobs whose arrival times to the system are as follows:
- Priorities are inverse of their value. Thus, a process with priority 1 is prioritized over another with priority 2 or 3.
- You are asked to fill in the following tables in the following cases:
  - a) Round-robin scheduling policy with slice of 1
  - b) Round-robin scheduling policy with slice of 4
  - c) SJF (Shortest Job First) scheduling Policy (Non-Expulsive)
- NOTE: If the execution slice of a process ends at the same instant that a new process arrives on the system, then the new process is placed in the ready-to-run queue before the process that expires the slice.

3. Statement proposed

- Draw the cronogram of the following processes.

PROCESSES	ARRIVAL	EXECUTION
P1	0	1ms CPU + 6ms E/S + 1ms CPU
P2	1	3 ms CPU
P3	3	5ms CPU + 3ms E/S + 1ms CPU
P4	3	3 ms CPU

- b) Indicate for each process their time of stay in the system and the penalty time suffered by each one of them.
- c) What is the worst treated process?

3. Solution.

A)

Proces	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
P1	CPU	E/S	E/S	E/S	E/S	E/S	E/S	CPU								
P2		CPU	CPU						CPU							
P3				CPU	CPU					CPU	CPU	CPU	E/S	E/S	E/S	CPU
P4						CPU	CPU						CPU			

B)

Process	Stay	Penalty
P1	8	0
P2	8	5
P3	13	4
P4	10	7

c) The worst treated process is process 4 since it takes 10 periods to finish its execution, when it only has 3 execution periods, it suffers 7 penalty periods.