

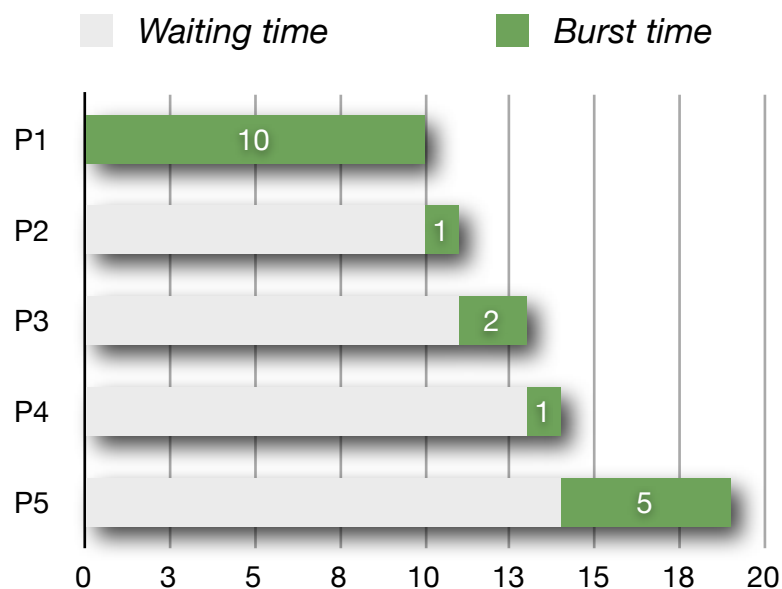
Tarea #2

I. Considere los siguientes procesos, junto con la longitud del CPU burst. Los procesos arriban en el orden P1,P2,P3,P4,P5 todos en el tiempo 0:

Proceso	Burst Time	Prioridad
P1	10	3
P2	1	1
P3	2	3
P4	1	4
P5	5	2

1. Algoritmo FCFS: first-come, first-served

a) Diagrama de Gantt

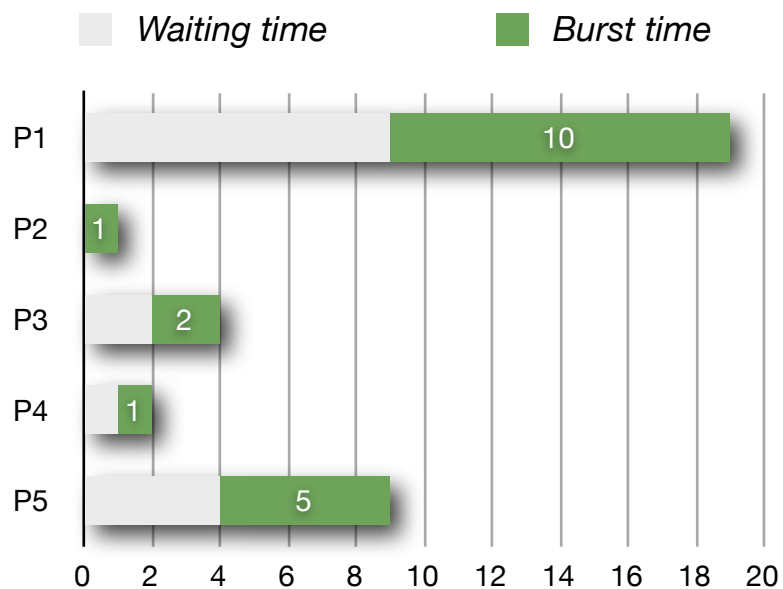


b) *Turnaround time y waiting time* de cada proceso

Proceso	<i>Turnaround time</i>	<i>Waiting time</i>
P1	10	0
P2	11	10
P3	13	11
P4	14	13
P5	19	14
Promedio	13.4	9.6

2. Algoritmo SJW: Shortest-job-first

A. Diagrama de Gantt



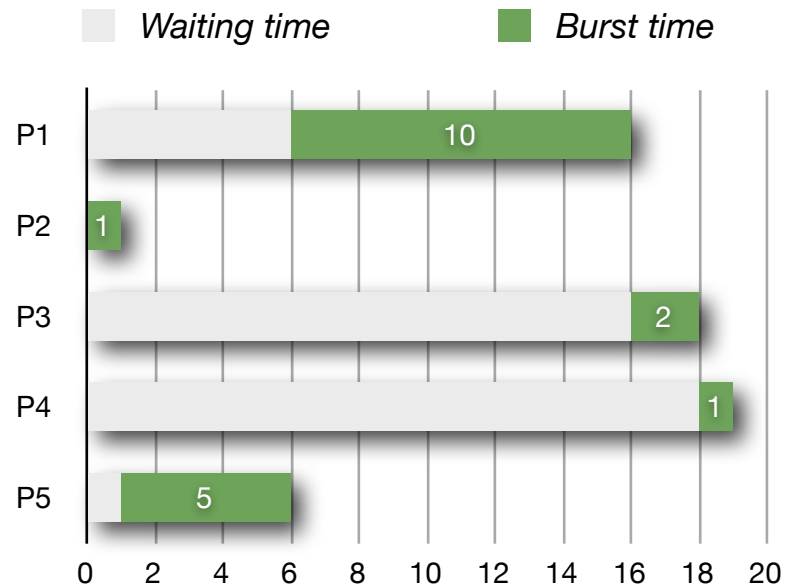
b) *Turnaround time y waiting time* de cada proceso

Proceso	<i>Turnaround time</i>	<i>Waiting time</i>
P1	19	9
P2	1	0
P3	4	2
P4	2	1

Proceso	Turnaround time	Waiting time
P5	9	4
Promedio	7	3.2

3. Algoritmo *Nonpreemptive priority* (menor número, mayor prioridad)

a) Diagrama de Gantt

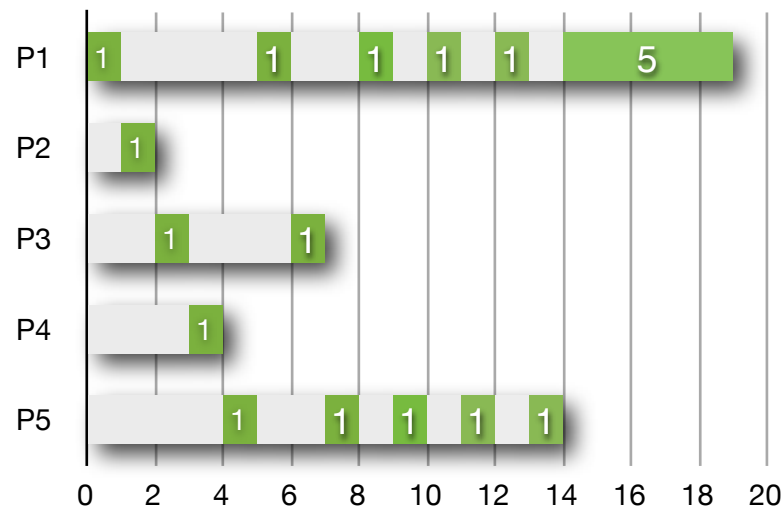
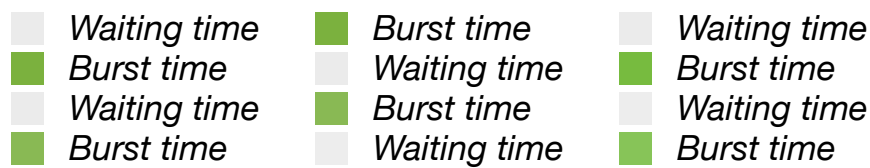


b) Turnaround time y waiting time de cada proceso

Proceso	Turnaround time	Waiting time
P1	16	6
P2	1	0
P3	18	16
P4	19	18
P5	6	1
Promedio	12	8.2

4. Algoritmo *Round robin* con *quantum=1*

a) Diagrama de Gantt



b) *Turnaround time* y *waiting time* de cada proceso

Proceso	<i>Turnaround time</i>	<i>Waiting time</i>
P1	19	9
P2	2	1
P3	7	5
P4	4	3
P5	13	8
Promedio	9	5.2

5. ¿Cuál de los algoritmos minimiza el *waiting time* para este conjunto de procesos?

El de *Shortest-job-first*, con un promedio de 3.2. Siguiendole el de *Round Robin*, con 5.2. El más tardado fue el de *First-come, first-served*.

6. ¿Qué algoritmos están propensos a *starvation* y por qué?

Prácticamente todos los que son *Nonpreemptive*, esos son los primeros tres: *First-come,first-serve*, *Shortest-job-first* y *Nonpreemptive priority*. Como son *Nonpreemptive*, una vez que el proceso es asignado al procesador, el procesador los ejecuta completamente, esperando a que estos terminen para que se le pueda asignar algún otro.