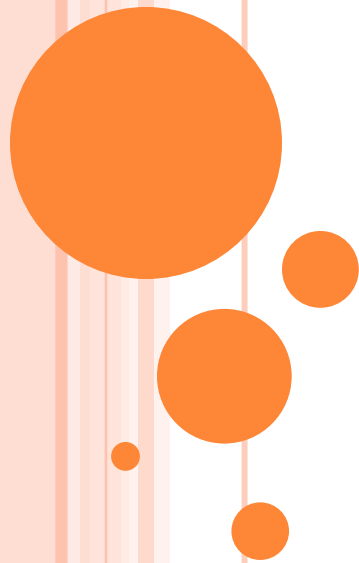


**SINCRONIZACIÓN PAQUETE 2**

# **ALGORITMOS DE ESPERA**



**DANIEL REINA HIDALGO**

# ALGORITMO DE ESPERA

Se encargan de gestionar el tiempo disponible del microprocesador entre todos los procesos listos para su ejecución.

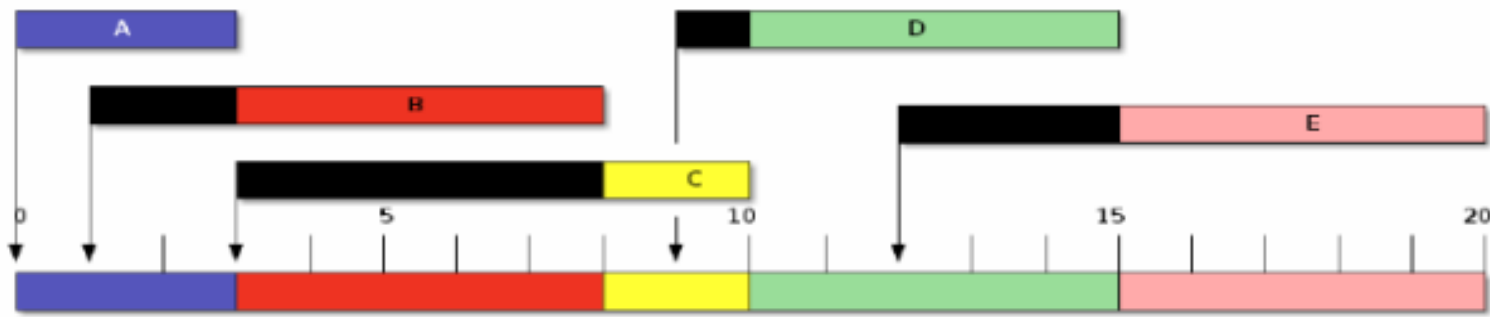
Existen dos tipos de algoritmos de espera:

- **Expropiativo:** También conocido como expulsivo, este tipo de algoritmo nos permite la expulsión de procesos llevándolos a la cola de espera, para ejecutar un nuevo proceso.
- **No expropiativo:** No nos permite la expulsión, por lo que un proceso nuevo no entrará hasta que termine el anterior.



# FCFS

- FCFS (FIRST-COME, FIRST-SERVED) es un algoritmo no expropiativo con una planificación de servicio por orden de llegada.

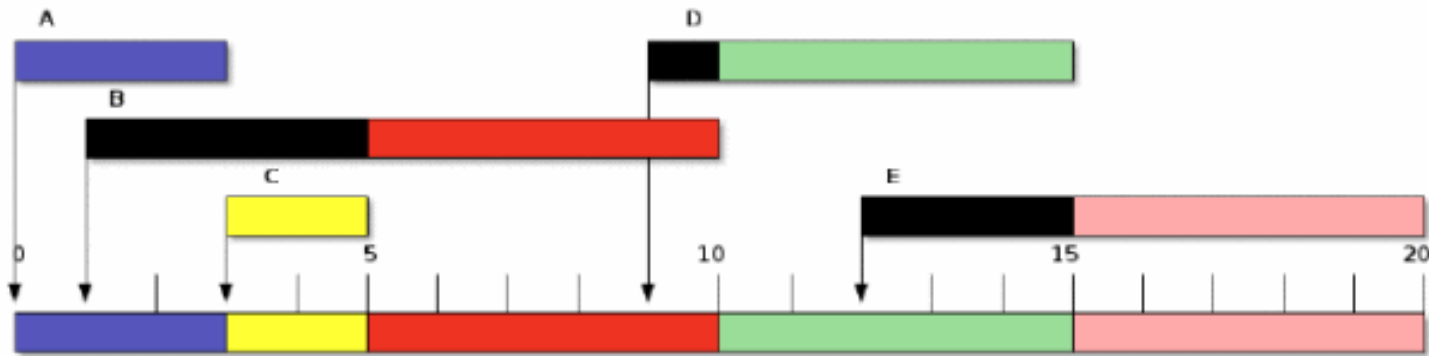


Proceso	Inicio	Fin	$T$	$E$	$P$
A	0	3	3	0	1
B	3	8	7	2	1.4
C	8	10	7	5	3.5
D	10	15	6	1	1.2
E	15	20	8	3	1.6
Promedio			6.2	2.2	1.74



# SJF

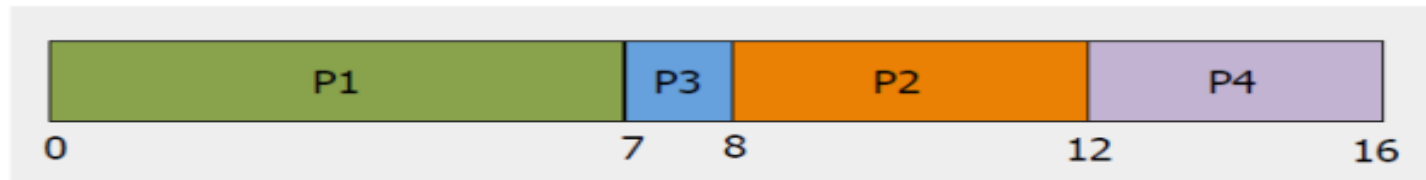
- **SJF** (Shortest Job First) seleccionará el proceso que requiera menor tiempo de ejecución. Se trata de un algoritmo no expropiativo y es un caso especial de planificación por prioridad.



# SJF

SJF (non preemptive)

Process	Arrival time	CPU burst
P1	0	7
P2	2	4
P3	4	1
P4	5	4



Mean waiting time:  $(0 + 6 + 3 + 7) / 4 = 4$

Proceso	Inicio	Fin	$T$	$E$	$P$
A	0	3	3	0	1.0
B	5	10	9	4	1.8
C	3	5	2	0	1.0
D	10	15	6	1	1.2
E	15	20	8	3	1.6
Promedio			5.6	1.6	1.32



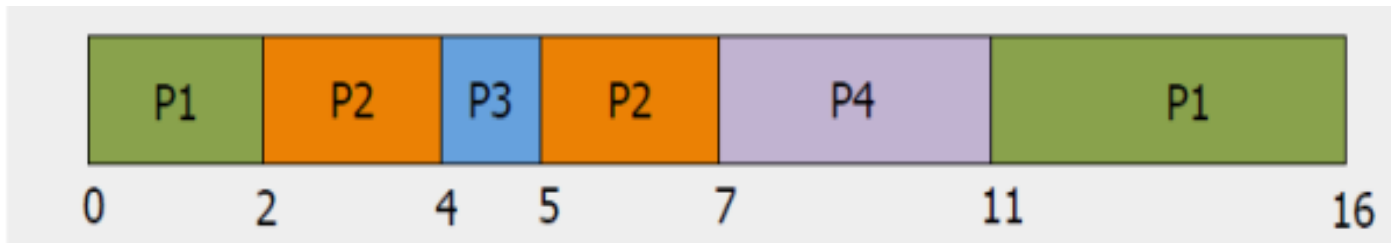
# SRTF

- **SRTF** (Shortest Remaining Time First) es la variante expropiativa del algoritmo SJF, en la que el proceso en CPU es desalojado si llega a la cola un proceso con duración más corta. En el caso que tengamos un proceso que requiera un tiempo de ejecución para finalizar igual que un proceso nuevo que entra existen dos soluciones: dar prioridad a los procesos nuevos sobre los procesos en ejecución o dar prioridad a los procesos en ejecución sobre los procesos nuevos.

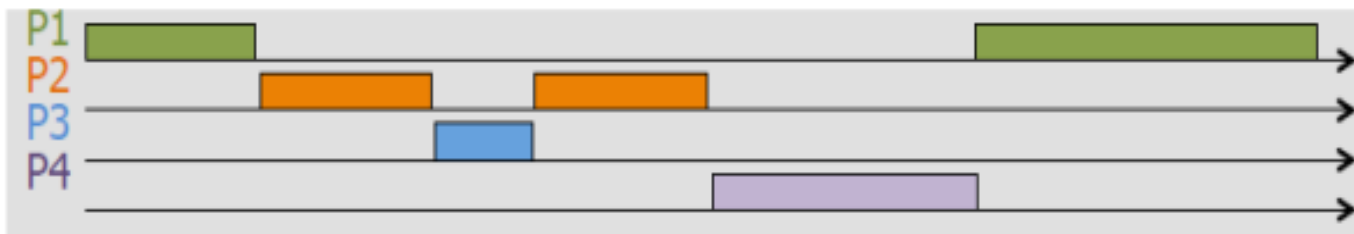


# SRTF

Process	Arrival time	CPU burst
P1	0	7
P2	2	4
P3	4	1
P4	5	4



Process time line



Mean waiting time:  $(9 + 1 + 0 + 2) / 4 = 3$



# RR

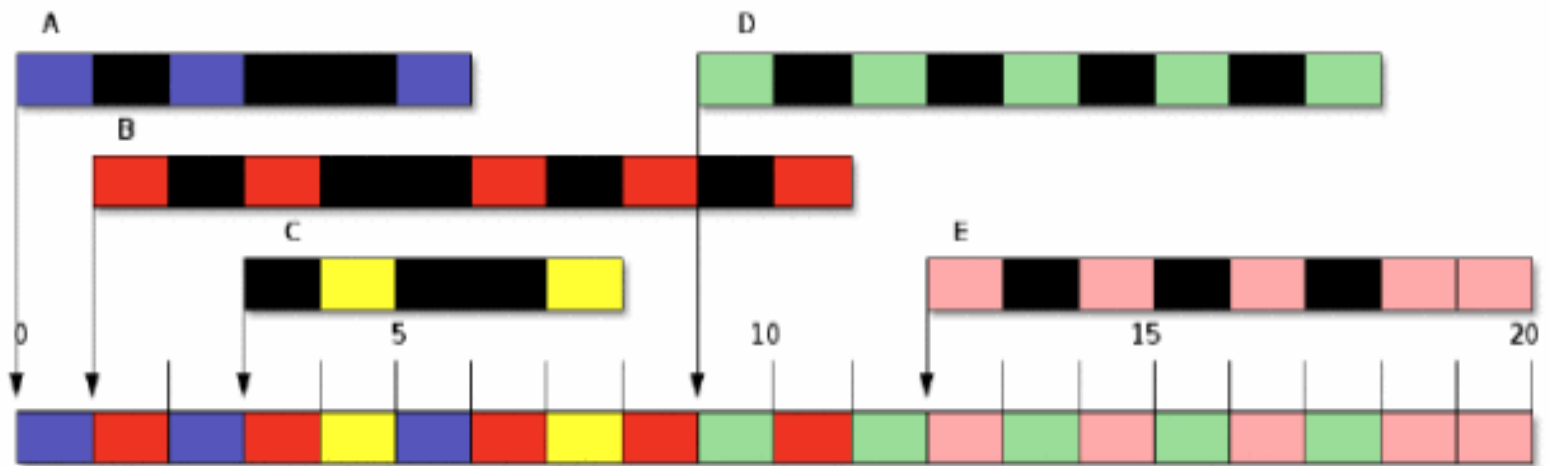
- **RR** (Round Robin) cada proceso dispone de un quantum (unidad de tiempo de CPU) de tiempo máximo, de tal forma que a cada proceso se le asigna un quantum de forma secuencial. La selección entre los procesos activos se gestiona según una cola FIFO, si cuando expira el quantum de tiempo el proceso continúa en CPU, el planificador lo desaloja y lo ingresa al final de la cola de preparados.

Por tanto, este algoritmo es expropiativo y no emplea prioridades.





# RR



Proceso	Inicio	Fin	$T$	$E$	$P$
A	0	6	6	3	2.0
B	1	11	10	5	2.0
C	4	8	5	3	2.5
D	9	18	9	4	1.8
E	12	20	8	3	1.6
Promedio			7.6	3.6	1.98



# COMPARACIÓN DE ALGORITMOS

- **FCFS** es el algoritmo más simple y fácil de implementar, apto para multitarea cooperativa. En contra, los tiempos de espera son bastante largos, los procesos largos pueden monopolizar la CPU durante mucho tiempo generando tiempos de espera mayores de los que serían deseables.
- **SJF** minimiza el tiempo de espera medio y favorece a los procesos cortos. El problema es que procesos muy largos puedan quedar siempre bloqueados por procesos más cortos (riesgo de inanición).



# COMPARACIÓN DE ALGORITMOS

- **SRTF** optimiza la media del tiempo de espera respecto SJF (mantiene mejores promedios porque los procesos cortos salen antes), pero acarrea los mismos problemas y sumando la dificultad de predecir la duración de la siguiente ráfaga de CPU.
- **RR** es adecuado para implementar tiempo compartido. Si el quantum es muy grande (más grande que el mayor tiempo de CPU de los procesos), el algoritmo se comporta como FCFS ya que los procesos terminan antes de que termine el quantum y si es muy pequeño provocarían constantemente cambios de contexto, disminuyendo el rendimiento.



## REFERENCIAS

- <https://marialopezweb.wordpress.com/2015/11/21/algoritmos-de-planificacion-de-procesos/>
- <https://emiliosedanogijon.wordpress.com/2014/09/16/algoritmos-de-planificacion-fcfs-sjf-srtf-round-robind/>
- <http://jmoral.es/blog/planificacion-procesos>
- <https://es.slideshare.net/FernandoJMoralesOldenburg/algoritmos-de-planificacin-de-procesos-en-sistemas-operativos>

