

Boletín de Ejercicios de Procesos

2º curso del Grado en Informática

11 de septiembre de 2023

Consideraciones iniciales

De cara a la realización de los ejercicios, habrá que rellenar un diagrama de Gantt, como el que se incluye a continuación, en el que se mostrará, para cada unidad de tiempo (u.t.) el estado en el que se encuentra cada proceso de entre listo (L), en ejecución (E) o bloqueado (B). Para elaborarlo, haremos las siguiente suposiciones:

- El tiempo de duración de las ráfagas de ejecución y entrada/salida de cada proceso se expresará en unidades de tiempo completas y supondremos que cada proceso siempre va a mantener su estado durante toda la unidad de tiempo.
- La unidad de tiempo X comienza en el instante X y termina en el instante X+1.
- Al comienzo de cada unidad de tiempo ocurre un *posible instante de planificación*, que por simplicidad suponemos que tiene duración 0, que es cuando el planificador del sistema operativo puede intervenir, y por lo tanto, decidir, a partir del algoritmo de planificación implementado, el estado en el que quedará cada proceso durante la unidad de tiempo correspondiente.
- Dependiendo del algoritmo de planificación implementado, habrá unidades de tiempo que tengan asociadas instantes de planificación y otras que no. Por ejemplo, cuando se utiliza el algoritmo FCFS y a un proceso se le otorga la CPU, el planificador no volverá a ser llamado (y, por lo tanto, no ocurrirá otro instante de planificación) hasta la unidad de tiempo siguiente a la que acaba la ráfaga de CPU del proceso.
- El tiempo de retorno de un proceso lo calcularemos contando el número de unidades de tiempo desde que llegó hasta la última unidad de tiempo en estado E (ambas inclusive).
- El tiempo de espera de un proceso lo calcularemos contando el número de unidades de tiempo que el proceso está en estado listo (L).

u.t.	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1	L→E	E																							
P_2	-	L	L	L	L	L→E																			
P_3	L	L	L→E	E	E																				

Así pues, y para el ejemplo que se muestra, asumiendo un algoritmo de planificación FCFS, vemos que los instantes de planificación ocurrirían al comienzo de las unidades de tiempo 0, 2 y 5. Más concretamente, vemos cómo al comienzo de la unidad de tiempo 0, el planificador encuentra dos procesos en la cola de listos, P_1 y P_3 , de entre los cuales, en el instante de planificación 0 (que, recordemos, tiene duración 0 y ocurre al comienzo de la unidad de tiempo correspondiente) el planificador decide otorgar la CPU a P_1 . En el instante de planificación que ocurre al comienzo de la unidad de tiempo 2, el planificador se encuentra con que hay dos procesos en la cola de listos (P_2 y P_3), pero que P_3 llegó antes que P_2 , con lo que dará la CPU al primero. Por último, una vez que P_3 ha completado su ráfaga de CPU de 3 unidades de tiempo, el planificador otorgará la CPU

al único proceso que queda (P_2) en el instante de planificación que ocurre al comienzo de la unidad de tiempo 5.

Tal y como podemos ver, el tiempo de respuesta de P_1 será de 2 unidades de tiempo, y su tiempo de espera de 0 unidades de tiempo. Para P_2 , los tiempos de respuesta y de espera serán de 5 y 4 unidades de tiempo, respectivamente, mientras que para P_3 , serán de 5 y 2, respectivamente.

Enunciados

1. En un sistema se ejecutan tres procesos que llegan todos en el instante 0. Todo proceso muestra un comportamiento de ráfagas de CPU y E/S alternas, como puede verse en la tabla donde los números indican tiempos expresados en tics de reloj. Un diagrama de Gantt sirve para mostrar en cada momento en qué estado se encuentra cada proceso (E - Ejecución, B - Bloqueado y L - en cola de listos). El tiempo en dicho diagrama indica el instante en que comienza el tic representado por la celda.

Proceso	CPU	E/S	CPU
1	3	2	2
2	6	2	1
3	1	4	4

Con estos datos se pide:

- a) Rellena el siguiente cronograma indicando el estado de cada proceso usando un planificador **FCFS**. Recuerda que es no apropiativo. Siempre que dos procesos entren a la vez en la cola de listos, introduciremos primero el que menor número de proceso tenga. Calcula el tiempo promedio de retorno y de espera de los procesos.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- b) Rellena el siguiente cronograma indicando el estado de cada proceso usando un planificador **SJF**. Recuerda que es **no apropiativo**. En este caso inicial, supón que el planificador es una especie de adivino y conoce la duración de cada ráfaga de CPU. Siempre que dos procesos entren a la vez en la cola de listos, introduciremos primero el que menor número de proceso tenga. Calcula el tiempo promedio de retorno y de espera de los procesos.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- c) Rellena el siguiente cronograma indicando el estado de cada proceso usando un planificador **SJF**. Recuerda que es **no apropiativo**. Ahora nos vamos a enfrentar al problema real donde el planificador no conoce la duración de cada ráfaga de CPU. Piensa que cuando le damos la CPU a un proceso, no sabemos cuándo querrá bloquearse, así que nunca conocemos la duración de la ráfaga de CPU. Tendremos que utilizar estimaciones para poder decidir. Supón que la estimación inicial de las ráfagas es de 4 tics y el coeficiente de credibilidad $\alpha = 0,5$. Siempre que dos procesos entren a la vez en la cola de

listos, introduciremos primero el que menor número de proceso tenga. Calcula el tiempo promedio de retorno y de espera de los procesos.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- d) Rellena el siguiente cronograma indicando el estado de cada proceso usando un planificador **SRTF** (menor tiempo restante primero). Recuerda que es **apropiativo**. En este caso inicial, supón que el planificador es una especie de adivino y conoce la duración de cada ráfaga de CPU. Siempre que dos procesos entren a la vez en la cola de listos, introduciremos primero el que menor número de proceso tenga. Calcula el tiempo promedio de retorno y de espera de los procesos.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- e) Rellena el siguiente cronograma indicando el estado de cada proceso usando un planificador **SRTF**. Recuerda que es **apropiativo**. Ahora nos vamos a enfrentar al problema real donde el planificador no conoce la duración de cada ráfaga de CPU. Piensa que cuando le damos la CPU a un proceso, no sabemos cuándo querrá bloquearse, así que nunca conocemos la duración de la ráfaga de CPU. Tendremos que utilizar estimaciones para poder decidir. Supón que la estimación inicial de las ráfagas es de 4 tics y el coeficiente de credibilidad $\alpha = 0,5$. Al trabajar con una estimación, iremos descontando de la estimación que tenemos, los tics que haya consumido de esa ráfaga, y saturaremos en 0. Por ejemplo, si tenemos una estimación de 4 tics y se han consumido 3, nuestra estimación para el proceso en ejecución será de 1 tic. Si agota los 4 tics nos quedaremos con una estimación de 0 hasta que acabe la ráfaga y podamos recalcular la estimación. En estas condiciones solamente nos puede quitar la CPU un proceso que entre en la cola de listos (pensadlo). Siempre que dos procesos entren a la vez en la cola de listos, introduciremos primero el que menor número de proceso tenga. Calcula el tiempo promedio de retorno y de espera de los procesos.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- f) Rellena el siguiente cronograma indicando el estado de cada proceso usando un planificador **Round-Robin** con quantum de 4 tics. Recuerda que es **apropiativo**. Cuando un proceso entra a la cola de listos, recibe un nuevo quantum y siempre que dos procesos entren a la vez en la cola de listos, introduciremos primero el que menor número de proceso tenga. Calcula el tiempo promedio de retorno y de espera de los procesos.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- g) Rellena el siguiente cronograma indicando el estado de cada proceso usando un planificador **Round-Robin** con quantum de 2 tics. Recuerda que es **apropiativo**. Cuando un proceso entra a la cola de listos, recibe un nuevo quantum y siempre que dos procesos entren a la vez en la cola de listos, introduciremos primero el que menor número de proceso tenga. Calcula el tiempo promedio de retorno y de espera de los procesos.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- h) Por último rellena el siguiente cronograma indicando el estado de cada proceso usando un planificador apropiativo **de 2 colas con realimentación**. La cola 0 es la de mayor prioridad y la 1 la de menor. La cola 0 utiliza un planificador por Round-Robin con quantum de 2 tics y la cola 1 otro con quantum de 3 tics. Si se está ejecutando un proceso de la cola 1, cualquier proceso que entrara en la cola 0 le quitaría la CPU. Siempre que dos procesos entren a la vez en la misma cola, introduciremos primero el que menor número de proceso tenga. Las condiciones de subida y bajada para este ejercicio son las siguientes:

- Todo proceso comienza su ejecución en la cola 0.
- Un proceso sube a la cola 0 si se bloquea sin consumir totalmente el quantum asignado.
- Un proceso baja a la cola 1 en el momento en que consuma totalmente el quantum asignado sin que nadie le quite la CPU. Recuerda que en Round-Robin, cada vez que un proceso entra a la cola de listos recibe un quantum completo.

Calcula el tiempo promedio de retorno y de espera de los procesos.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

2. Un sistema operativo utiliza un algoritmo de planificación apropiativo formado por dos colas: la cola 0 es la de mayor prioridad y se planifica por Round-Robin con quantum de 2 tics, la cola 1 utiliza también Round-Robin pero con quantum de 3 tics. Cuando se inicia un proceso ingresa en la cola 0. Un proceso baja a la cola 1 en el momento en que consuma totalmente el quantum asignado. Un proceso sube a la cola 0 si se bloquea sin consumir su quantum. Asume que un proceso recibe un nuevo quantum en cualquier de las siguientes 4 situaciones: (i) tras consumir el quantum en curso, (ii) al desbloquearse, (iii) cuando cambia de cola, o (iv) si otro proceso más prioritario le quita la CPU. Siempre que dos procesos entren a la vez en la misma cola, introduciremos primero el que menor número de proceso tenga. Utiliza en los estados E y L un subíndice indicando la cola en la que se encuentra el proceso (p.e. E_0 , L_1).

En este sistema se ejecutan los siguientes procesos (tiempo en tics) con los tiempos de llegada indicados.

Proceso	llegada	CPU	E/S	CPU
1	0	2	2	3
2	1	6	4	1
3	2	1	4	3

Con estos datos se pide:

- a) Dibuja el diagrama de Gantt en el cronograma que sigue. Marca lo que está ocurriendo a cada proceso, para ello utiliza las letras: E (en ejecución), B (bloqueado), L (en la cola de listos). **Recuerda usar un subíndice** para indicar la cola en la que se encuentra el proceso. El tiempo indica el instante en el que comienza el tic representado por la celda.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- b) Supongamos que queremos utilizar una versión un poco más elaborada del Round-Robin usado por el planificador anterior. Hasta este momento cada vez que un proceso entra en la cola de listos recibe un quantum nuevo independientemente de su historia. Ahora, un proceso solamente recibe un quantum nuevo tras consumir el actual a pesar de que haya sido interrumpido por otro proceso más prioritario, o por entrar a la cola de listos tras haber estado bloqueado o haber cambiado de cola. Si un proceso más prioritario le quita la CPU, al recibirla de nuevo tendrá el resto del quantum que le quedaba por consumir. Las reglas de ascenso y descenso quedan como siguen: un proceso baja a la cola 1 si consume totalmente su quantum sin bloquearse, y sube a la cola 0 si se bloquea antes de consumir su quantum.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- c) Calcula el tiempo promedio de retorno y el tiempo promedio de espera de los procesos en cada uno de los anteriores supuestos.
3. Un sistema operativo utiliza un algoritmo de planificación apropiativo de múltiples colas con realimentación formado por tres colas: La cola 0 para procesos del sistema sumamente importantes, y las colas 1 y 2 para procesos interactivos. La cola 0 es la de mayor prioridad y usa un algoritmo FIFO; la cola 1 utiliza Round-Robin pero con quantum de 2 tics y finalmente la cola 2 utiliza Round-Robin pero con quantum de 3 tics. En este ejercicio un proceso recibe un nuevo quantum tras consumir el quantum en curso, al desbloquearse, o al cambiar de cola. Si otro proceso más prioritario le quita la CPU, eso no implica nuevo quantum sino que continúa con lo que le quedara del antiguo. El administrador indica en qué cola entra cada proceso. Un proceso de la cola 0 siempre permanece en ella. Un proceso de la cola 1 baja a la cola 2 **en el momento en que consume su quantum sin bloquearse**. Un proceso sube a la cola 1 si se **bloquea sin consumir su quantum, aunque acabe de llegar a la cola 2**. Utiliza en los estados E y L un subíndice indicando la cola en la que se encuentra el proceso.

En este sistema se ejecutan los siguientes procesos (tiempo en tics) con los tiempos de llegada indicados.

Proceso	cola	llegada	CPU	E/S	CPU	E/S	CPU
1	1	0	4	3	1		
2	1	1	1	2	1	2	3
3	1	4	2	2	3		
4	0	0	–	10	3	8	

Con estos datos se pide:

- a) Dibuja el diagrama de Gantt en el cronograma que sigue. Marca lo que está ocurriendo a cada proceso, para ello utiliza las letras: E (en ejecución), B (bloqueado), L (en la cola de listos). **Recuerda usar un subíndice** para indicar la cola en la que se encuentra el proceso. El tiempo indica el instante en el que comienza el tic representado por la celda.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20
P_1																					
P_2																					
P_3																					
P_4																					

- b) Calcula el tiempo promedio de retorno y el tiempo promedio de espera de los procesos interactivos.
4. Tenemos un sistema operativo con un planificador de procesos que sigue un algoritmo apropiativo con múltiples colas de prioridad, compuesto por dos colas sin posibilidad de que los procesos cambien de cola. La cola 1 es la de más prioridad y utiliza un algoritmo round-robin con quantum 2 unidades de tiempo. La cola 2 usa un algoritmo SRTF con estimación inicial de 2 unidades de tiempo y un coeficiente de credibilidad $\alpha = 0,5$. Tenemos 4 procesos, donde los dos primeros (P_1 y P_2) se están ejecutando en la cola 1 y los otros dos (P_3 y P_4) se están ejecutando en la cola 2. Las ráfagas de CPU y E/S de cada proceso se pueden ver en la siguiente tabla, donde todos los procesos entran a la vez en el instante 0:

Cola	Proceso	CPU	E/S	CPU	E/S	CPU	E/S	CPU
1	1	1	3	1	5	2	3	1
1	2	1	4	1	5	1	5	1
2	3	4	2	3				
2	4	1	5	2				

Se pide responder a las siguientes cuestiones:

- a) Rellena el cronograma que sigue, donde el tiempo indica el instante en el que comienza la unidad de tiempo representada por la celda. Marca el estado de cada proceso utilizando las letras E (en ejecución), B (bloqueado) y L (en la cola de listos), incluyendo en SRTF, como un subíndice, la estimación al comienzo de cada tic de reloj. En caso de empate, coge la CPU el proceso de menor número.
- b) Calcula los tiempos medios de espera y retorno de los procesos de cada cola por separado. Calcula el tiempo medio de espera y retorno total.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									
P_4																									

5. Queremos ejecutar 4 procesos con las ráfagas de CPU y E/S indicadas en la tabla siguiente:

Proceso	T. llegada	CPU	E/S	CPU	E/S	CPU
1	0	4	2	5		
2	0	1	3	1	2	1
3	4	4	3	1		
4	4	1	3	1		

Supongamos que tenemos un sistema operativo con un planificador apropiativo de múltiples niveles con realimentación compuesto por dos colas (la cola 0, de más prioridad y quantum de 2 u.t., y la cola 1 de menos prioridad y quantum de 4 u.t.). Todos los procesos comienzan en la cola 0, pero si consumen su quantum sin bloquearse, se pasan a la cola 1, sin posibilidad de volver. Toda la E/S que se produce va al mismo disco, y el planificador de disco sigue la política FCFS, lo que hace que hasta que no se termine de servir una petición, no comienza la siguiente. Esto implica que si un proceso solicita E/S al disco con una duración de 4 u.t., pero ya hay una petición en curso a la que le quedan 2 u.t., ese proceso permanecerá bloqueado 6 u.t. (2 hasta que termine la que hay en curso, y 4 que le corresponden).

a) A continuación se muestra un cronograma parcialmente completado. Pon en cada una de las letras que aparecen un subíndice indicando la cola en la que se encuentra el proceso en ese momento. Termina de completar el cronograma con la misma notación. En el caso de que dos procesos lleguen a la misma cola de listos a la vez, se considerará que entra primero el que tiene número de proceso menor.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1	E	E	L	E	L	L	L	L	E	B															
P_2	L	L	E	B	B	L	L	E	B	B															
P_3	-	-	-	-	E	E	L	L	E																
P_4	-	-	-	-	L	L	E	B	B	B															

b) Calcula los tiempos medios de espera y retorno de los procesos.

6. En un sistema se ejecutan los siguientes procesos (tiempo en tics) con los tiempos de llegada y de ráfagas de CPU y E/S indicados:

Proceso	Llegada	CPU	E/S	CPU	E/S	CPU
1	0	6	1	3	1	1
2	0	1	2	2	2	1
3	8	3	1	1	1	1

Usando los planificadores indicados en los siguientes apartados, dibuja sendos diagramas de Gantt en los cronogramas que siguen, donde el tiempo indica el instante en el que comienza el tic representado por la celda. Marca el estado de cada proceso utilizando las letras E (en ejecución), B (bloqueado) y L (en la cola de listos).

En ambos casos, se supone que el planificador no conoce la duración de cada ráfaga de CPU, con lo que tendrá que manejar estimaciones para poder decidir. Supón que la estimación inicial de las ráfagas es de 2 tics y el coeficiente de credibilidad $\alpha = 0,5$. En caso de empate, supón que coge la CPU el proceso de menor número.

a) Se usa un planificador **SJF**.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

b) Se usa un planificador **SRTF**.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

7. Un sistema operativo utiliza un algoritmo de planificación de procesos apropiativo de múltiples niveles sin realimentación compuesto por dos colas (cola 0 y cola 1), con las siguientes características:

- La planificación entre colas se realiza mediante el esquema Round-Robin con un quantum de 4 u.t.
- La planificación dentro de la cola 0 se lleva a cabo mediante un esquema de prioridades apropiativo, donde la prioridad de cada proceso es un número entre 0 y 3 (mayor número, mayor prioridad) que se calcula antes del comienzo de cada ráfaga de CPU, cuando el proceso entra en la cola 0 (es decir, cuando llega al estado Listo antes de iniciar una ráfaga de CPU). En particular, la prioridad de un proceso se calcula como la diferencia entre 4 y el número de tics de su ráfaga de CPU anterior, y se mantiene durante toda la ráfaga de CPU. La primera vez que un proceso llega a la cola 0, recibe como prioridad su número de proceso.
- Para la cola 1, sin embargo, se utiliza el esquema SRTF con estimaciones para poder decidir. Supón que la estimación inicial de las ráfagas es de 2 tics y el coeficiente de credibilidad $\alpha = 0,5$.
- En caso de empate, supón que se le da la CPU al proceso de menor número.

En este sistema se ejecutan los siguientes procesos, donde todos los tiempos vienen dados en tics de reloj:

Proceso	Cola	Llegada	CPU	E/S	CPU	E/S	CPU
1	0	0	2	1	2	2	1
2	0	0	3	2	2	1	1
3	1	3	3	2	2	—	—
4	1	2	2	1	3	—	—

a) Dibuja el diagrama de Gantt en el cronograma que sigue, donde el tiempo indica el instante en el que comienza el tic representado por la celda. Para ello asume solamente la presencia de los procesos 1 y 2. Marca el estado de cada proceso utilizando las letras E (en ejecución), B (bloqueado) y L (cola de listos). Añade a los estados E y L un subíndice (p.e., E_1 , L_2) para indicar la prioridad que tiene el proceso.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									

- b) Dibuja el diagrama de Gantt en el cronograma que sigue, asumiendo ahora la presencia únicamente de los procesos 3 y 4. Marca el estado de cada proceso utilizando las letras E (en ejecución), B (bloqueado) y L (en la cola de listos). Añade a los estados E y L un subíndice (p.e., E_1 , L_2) para indicar la la estimación al comienzo de cada tic de reloj.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_3																									
P_4																									

- c) Considerando ahora los 4 procesos, completa el diagrama de Gantt en el cronograma que sigue. Marca el estado de cada proceso utilizando las letras E (en ejecución), B (bloqueado) y L (en la cola de listos). Añade a los estados E y L un subíndice (p.e., $E_{0,1}$, $L_{1,2}$) para indicar la cola en la que está el proceso y la prioridad que tiene (si está en la cola 0) o la estimación al comienzo de cada tic de reloj (si está en la cola 1).

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									
P_4																									

- d) Calcula también los tiempos medios de respuesta y espera de los procesos para el apartado anterior.
8. Un sistema operativo utiliza un algoritmo de planificación por prioridades no apropiativo. A mayor número, mayor es la prioridad de un proceso. Para evitar la inanición de los procesos de menor prioridad, la prioridad de un proceso listo se incrementa en 1 por cada unidad de tiempo que pasa en la cola de listos, si bien la prioridad del proceso vuelve a su valor inicial una vez haya cogido la CPU.

En este sistema, se ejecutan los siguientes procesos (tiempo en tics) con las prioridades iniciales y tiempos de llegada indicados:

Proceso	Prioridad	Llegada	CPU	E/S	CPU	E/S	CPU
1	0	0	2	3	3	1	1
2	3	1	4	2	3	1	1
3	4	4	2	2	2	2	2

Con estos datos se pide:

- a) Dibuja el diagrama de Gantt en el cronograma que sigue, donde el tiempo indica el instante en el que comienza el tic representado por la celda. Marca el estado de cada proceso utilizando las letras E (en ejecución), B (bloqueado) y L (en la cola de listos). **Recuerda usar un subíndice** en el caso de los procesos listos para indicar su prioridad actual. En caso de empate, supón que coge la CPU el proceso de menor número.
- b) Calcula el tiempo promedio de retorno y el tiempo promedio de espera de los procesos.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

9. Un sistema dispone de una CPU y dos dispositivos de entrada/salida: un disco y una impresora. Existen tres procesos que para ejecutarse necesitan emplear los distintos recursos del sistema tal y como aparece en esta tabla:

Proceso	Llegada	CPU	Disco	CPU	Impresora	CPU
P_1	0	2	1	1	2	2
P_2	1	3	1	2	1	1
P_3	2	1	2	2	1	2

De modo que, el proceso 1 necesita 2 unidades de tiempo la CPU, a continuación 1 unidad de tiempo el disco, luego 1 u.t. en CPU, luego 2 u.t. en la impresora y termina con 2 u.t. en CPU. El disco y la impresora son recursos que no pueden servir más de una petición a la vez, así que si hay varias tienen que encolarse y se planifican mediante un FCFS en este caso. Si varios procesos llegan a una cola al mismo tiempo, se considera que ha llegado antes el proceso con mayor número.

Utilizando un algoritmo de planificación round-robin con quantum de 2 u.t., rellena el cronograma que se muestra a continuación indicando con D que el proceso está bloqueado esperando disco y con I que está bloqueado esperando impresora. Calcula el tiempo medio de espera y el tiempo medio de retorno.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

10. Tenemos tres procesos con las ráfagas de CPU y E/S indicadas en la tabla. Estamos ejecutándolos en un sistema multiprocesador con 2 CPUs (CPU_0 y CPU_1), y planificación round-robin con quantum de 2 unidades de tiempo. Cuando una CPU se queda disponible, llama al planificador para que le asigne un proceso de la cola de listos. En el caso de que dos procesos lleguen a la cola de listos a la vez, se supone que llega primero el de número menor.

Proceso	CPU	E/S	CPU	E/S	CPU
P1	2	6	3	3	1
P2	2	4	4	2	1
P3	2	3	3	3	2

- a) Dibuja en un diagrama la ejecución de los tres procesos en este sistema, indicando con una L si el proceso está en la cola de listos, con una B si está bloqueado y con una E_n si está en ejecución en ese momento en la CPU n. Si las dos CPUs quedan libres a la vez, se supone que se asignan procesos en el orden CPU_0 y CPU_1 .

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- b) Dibuja en un diagrama la ejecución de los mismos procesos si solamente tenemos una CPU.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

- c) Dibuja en un diagrama la ejecución de los mismos procesos en 1 CPU pero utilizando el algoritmo de planificación “el más corto primero” (SJF) interactivo. Supón que la estimación inicial de tiempo es 3, 2 y 1 para P_1 , P_2 y P_3 respectivamente, y que el factor alfa es 0,5. En caso de empate toma el proceso con número menor.

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1																									
P_2																									
P_3																									

Soluciones a ejercicios seleccionados

■ Ejercicio 1:

a) Cronograma FCFS:

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P1	E	E	E	B	B	L	L	L	L	L	E	E													
P2	L	L	L	E	E	E	E	E	E	B	B	L	E												
P3	L	L	L	L	L	L	L	L	L	E	B	B	B	B	E	E	E	E							

Y los tiempos medios de retorno y de espera quedan: $T_r = \frac{12+13+18}{3} = 14,33$ tics y $T_e = \frac{5+4+9}{3} = 6$ tics.

b) Cronograma SJF oráculo:

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P1	L ₃	E ₃	E ₃	E ₃	B ₂	B ₂	L ₂	L ₂	L ₂	L ₂	E ₂	E ₂													
P2	L ₆	L ₆	L ₆	L ₆	E ₆	E ₆	E ₆	E ₆	E ₆	E ₆	B ₁	B ₁	E ₁												
P3	E ₁	B ₄	B ₄	B ₄	B ₄	L ₄	L ₄	L ₄	L ₄	L ₄	L ₄	L ₄	L ₄	E ₄	E ₄	E ₄	E ₄								

Donde el subíndice en E indica la estimación para la ráfaga actual de CPU de ese proceso, y en L y B, la estimación para la siguiente ráfaga de CPU.

Y los tiempos medios de retorno y de espera quedan: $T_r = \frac{12+13+17}{3} = 14$ tics y $T_e = \frac{5+4+8}{3} = 5,67$ tics

c) Cronograma SJF estimación:

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P1	E ₄	E ₄	E ₄	B _{3,5}	B _{3,5}	L _{3,5}	L _{3,5}	L _{3,5}	L _{3,5}	E _{3,5}	E _{3,5}														
P2	L ₄	L ₄	L ₄	E ₄	E ₄	E ₄	E ₄	E ₄	E ₄	B ₅	B ₅	L ₅	E ₅												
P3	L ₄	L ₄	L ₄	L ₄	L ₄	L ₄	L ₄	L ₄	L ₄	L ₄	L ₄	E ₄	B _{2,5}	B _{2,5}	B _{2,5}	B _{2,5}	E _{2,5}	E _{2,5}	E _{2,5}	E _{2,5}					

Donde el subíndice en E indica la estimación para la ráfaga actual de CPU de ese proceso, y en L y B, la estimación para la siguiente ráfaga de CPU.

Y los tiempos medios de retorno y de espera quedan: $T_r = \frac{11+13+20}{3} = 14,67$ tics y $T_e = \frac{4+4+11}{3} = 6,33$ tics.

d) Cronograma SRTF oráculo:

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P1	L ₃	E ₃	E ₂	E ₁	B ₂	B ₂	E ₂	E ₁																	
P2	L ₆	L ₆	L ₆	L ₆	E ₆	L ₅	L ₅	L ₅	L ₅	L ₅	L ₅	E ₅	E ₄	E ₃	E ₂	E ₁	B ₁	B ₁	E ₁						
P3	E ₁	B ₄	B ₄	B ₄	B ₄	E ₄	L ₃	L ₃	E ₃	E ₂	E ₁														

Donde el subíndice indica la duración de la siguiente ráfaga de CPU cuando va con L o con B, y cuando está en E, el número de tics restantes de la ráfaga de CPU correspondiente al comienzo del tic en el que aparece. Los tiempos medios de retorno y espera quedan: $T_r = \frac{8+19+11}{3} = 12,67$ tics y $T_e = \frac{1+10+2}{3} = 4,33$ tics.

e) Cronograma SRTF con estimación:

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P1	E_4	E_3	E_2	$B_{3,5}$	$B_{3,5}$	$L_{3,5}$	$L_{3,5}$	$L_{3,5}$	$L_{3,5}$	$E_{3,5}$	$E_{2,5}$														
P2	L_4	L_4	L_4	E_4	E_3	E_2	E_1	E_0	E_0	B_5	B_5	L_5	E_5												
P3	L_4	L_4	L_4	L_4	L_4	L_4	L_4	L_4	L_4	L_4	L_4	E_4	$B_{2,5}$	$B_{2,5}$	$B_{2,5}$	$B_{2,5}$	$E_{2,5}$	$E_{1,5}$	$E_{0,5}$	E_0					

Al tener todos la misma estimación inicial, entran en el orden que indica su número de proceso. E_4 indica que al comienzo del tic de ejecución quedan un total de 4 tics a la ráfaga (duración estimada), L_3 y B_3 indican que la estimación para la siguiente ráfaga de CPU de ese proceso es 3 tics de duración. Por ejemplo, en el tic 3, $B_{3,5}$ indica que, como la ráfaga ha durado realmente 3 tics, la nueva estimación para la siguiente ráfaga será de $\alpha \cdot 4 + (1 - \alpha) \cdot 3 = 0,5 \cdot 4 + 0,5 \cdot 3 = \frac{4+3}{2} = 3,5$ tics. En el tic 9, $E_{3,5}$ indica que la estimación para la ráfaga actual al comienzo del tic 9 es de 3,5 tics.

Y los tiempos medios de retorno y de espera quedan: $T_r = \frac{11+13+20}{3} = 14,67$ tics y $T_e = \frac{4+4+11}{3} = 6,33$ tics.

f) Cronograma Round-Robin con quantum de 4 tics:

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P1	E	E	E	B	B	L	L	L	E	E															
P2	L	L	L	E	E	E	E	L	L	L	E	E	B	B	L	L	E								
P3	L	L	L	L	L	L	L	E	B	B	B	B	E	E	E	E									

Y los tiempos medios de retorno y de espera quedan: $T_r = \frac{10+17+16}{3} = 14,33$ tics y $T_e = \frac{3+8+7}{3} = 6$ tics.

g) Cronograma Round-Robin con quantum de 2 tics:

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P1	E	E	L	L	L	E	B	B	E	E															
P2	L	L	E	E	L	L	E	E	L	L	E	E	B	B	E										
P3	L	L	L	L	E	B	B	B	B	L	L	L	E	E	L	E	E								

En el tic 8 llegan P_1 y P_2 a la vez a la cola de listos, por tanto entran en el orden de su número de proceso y se selecciona a P_1 para coger la CPU.

Y los tiempos medios de retorno y de espera quedan: $T_r = \frac{10+15+17}{3} = 14$ tics y $T_e = \frac{3+6+8}{3} = 5,67$ tics.

h) Cronograma 2 colas con realimentación:

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P1	E_0	E_0	L_1	L_1	L_1	E_1	B_0	B_0	E_0	E_0															
P2	L_0	L_0	E_0	E_0	L_1	L_1	E_1	E_1	L_1	L_1	L_1	L_1	E_1	E_1	B_0	B_0	E_0								
P3	L_0	L_0	L_0	L_0	E_0	B_0	B_0	B_0	B_0	L_0	E_0	E_0	L_1	L_1	E_1	E_1									

Donde el subíndice indica la cola en la que se encuentra el proceso.

Y los tiempos medios de retorno y de espera quedan: $T_r = \frac{10+17+16}{3} = 14,33$ tics y $T_e = \frac{3+8+7}{3} = 6$ tics.

■ Ejercicio 2:

a) El cronograma sería el siguiente:

Tic	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
P1	E_0	E_0	B	B	L_1	E_1	E_1	E_1											
P2	–	L_0	E_0	E_0	L_1	L_1	L_1	L_1	E_1	L_1	L_1	E_1	E_1	E_1	B	B	B	B	E_1
P3	–	–	L_0	L_0	E_0	B	B	B	B	E_0	E_0	L_1	L_1	L_1	E_1				

b) El cronograma sería el siguiente:

Tic	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
P1	E_0	E_0	B	B	L_1	E_1	E_1	E_1												
P2	–	L_0	E_0	E_0	L_1	L_1	L_1	L_1	E_1	L_1	L_1	E_1	E_1	L_1	E_1	B	B	B	B	E_0
P3	–	–	L_0	L_0	E_0	B	B	B	B	E_0	E_0	L_1	L_1	E_1						

c) Los tiempos medios de retorno y de espera para el primer apartado quedan: $T_r = \frac{8+18+13}{3} = 13$ tics y $T_e = \frac{1+7+5}{3} = 4,33$ tics.

Para el segundo apartado, el tiempo medio de espera será $T_e = \frac{1+8+4}{3} = 4,33$ tics, y el tiempo medio de retorno, $T_r = \frac{8+19+12}{3} = 13$ tics.

■ Ejercicio 3:

a) La solución del ejercicio se desarrolla en los siguientes apartados:

Tic	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
P1	E_1	E_1	L_2	E_2	L_2	L_2	L_2	E_2	B	B	B	L_1	L_1	L_1	E_1			
P2	–	L_1	E_1	B	B	L_1	E_1	B	B	E_1	L_1	L_1	L_1	E_1	L_2	L_2	L_2	E_2
P3	–	–	–	–	E_1	E_1	B	B	E_2	L_2	L_2	L_2	L_2	L_2	L_2	E_2	E_2	
P4	B	B	B	B	B	B	B	B	B	B	E_0	E_0	E_0	B	B	B	B	B

b) Tiempo medio de espera: $T_e = \frac{7+8+6}{3} = 7$ tics. Tiempo medio de retorno: $T_r = \frac{15+17+13}{3} = 15$ tics

■ Ejercicio 4:

a) El cronograma sería el siguiente:

Tic	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
P_1	E	B	B	B	E	B	B	B	B	B	E	E	B	B	B	E					
P_2	L	E	B	B	B	B	E	B	B	B	B	B	E	B	B	B	B	B	E		
P_3	L_2	L_2	E_2	E_1	L_0	E_0	L_0	E_0	B_3	B_3	L_3	L_3	L_3	E_3	L_2	L_2	L_2	E_2	L_1	E_1	
P_4	L_2	L_2	L_2	L_2	L_2	L_2	L_2	L_2	E_2	$B_{1,5}$	$B_{1,5}$	$B_{1,5}$	$B_{1,5}$	$B_{1,5}$	$E_{1,5}$	$L_{0,5}$	$E_{0,5}$				

b) Tiempo medio de retorno de la cola 1: $\frac{16+19}{2} = 17,5$ tics. Tiempo medio de retorno de la cola 2: $\frac{20+17}{2} = 18,5$ tics. Tiempo medio de espera de la cola 1: $\frac{0+1}{2} = 0,5$ tics. Tiempo medio de espera de la cola 2: $\frac{11+9}{2} = 10$ tics.

El tiempo medio de retorno total sería $\frac{16+19+20+17}{4} = 18$ tics. El tiempo medio de espera total sería: $\frac{0+1+11+9}{4} = 5,25$ tics.

■ Ejercicio 5:

Tic	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
P_1	E_0	E_0	L_1	E_1	L_1	L_1	L_1	E_1	B_1	B_1	B_1	B_1	B_1	B_1	E_1	E_1	E_1	E_1	L_1	E_1					
P_2	L_0	L_0	E_0	B_0	B_0	B_0	L_0	E_0	B_0	B_0	B_0	B_0	E_0												
P_3	–	–	–	–	E_0	E_0	L_1	L_1	L_1	E_1	L_1	E_1	B_1	B_1	B_1	B_1	B_1	L_1	E_1						
P_4	–	–	–	–	L_0	L_0	E_0	B_0	B_0	B_0	E_0														

a) El cronograma sería el siguiente.

b) El tiempo medio de retorno es $\frac{20+13+15+7}{4} = 13,75$ u.t. y el tiempo medio de espera es $\frac{6+3+5+2}{4} = 4$ u.t.