

EJERCICIOS TEMAS 1 Y 2 - SO1 02-03

SOLUCIONES

1. Enumere las diferencias entre un sistema por lotes multiprogramado y un sistema de tiempo compartido.
Un sistema de tiempo compartido es un sistema multiprogramado en el cual se permite la interacción del usuario con el ordenador. En los sistemas por lotes, la interacción la realiza exclusivamente el operador del sistema.
2. Defina el concepto de monitor residente y enumere las funciones que realiza en los sistemas por lotes.
Es un programa que se carga al arrancar la máquina y permanece en memoria continuamente. Se considera el precursor de los sistemas operativos. Sus funciones son: manejo de dispositivos, carga y secuenciamiento de trabajos e intérprete de tarjetas.
3. ¿Cual es el objetivo fundamental de la multiprogramación?
*Aumentar la utilización de la CPU.
Aprovechar los intervalos de tiempo que un proceso se encuentra suspendido para ejecutar otro proceso y mantener así la CPU ocupada.*
4. Cite tres atributos que se hereden de procesos padres a procesos hijos en UNIX.
La imagen de memoria, el UID, el GID, la tabla de descriptores de ficheros, etc.
5. ¿Cual es el mecanismo para eliminar o interrumpir la ejecución de procesos en UNIX?. Ponga un ejemplo.
*Las señales. Su acción por defecto es matar, aunque se les puede asociar un manejador.
Ejemplos de señales: Pulsar la tecla CTRL-C, una división por cero, la orden kill.*
6. Cite las dos características que crea que mejor definen las siguientes generaciones de sistemas operativos:
 - a) Sistemas por lotes
 - b) Sistemas de tiempo compartido
 - c) Sistemas operativos actuales para computadores personales.*A) Tarjetas perforadas, operador
B) Multiprogramación, interacción usuario-máquina
C) Sistemas de ventanas, multitarea*
7. Cite cuatro ejemplos de programas del sistema.
*copiar un fichero (Ej: cp en Unix)
editar un fichero (Ej: vi en Unix)
obtener la hora del sistema (Ej: date en Unix)
compilar un programa (Ej: cc en Unix)
entre otros muchos ejemplos.*
8. Definir llamada al sistema y proporcionar varios ejemplos de ellas.
*Una llamada al sistema es el mecanismo de invocación de un servicio del sistema operativo. El conjunto de llamadas al sistema constituye el juego de instrucciones del sistema operativo. Las llamadas al sistema normalmente se implementan como interrupciones software o traps y pueden invocarse desde un lenguaje de programación (no desde el intérprete de órdenes) a través de funciones de biblioteca.
Ejemplos: Crear un proceso (fork()), ejecutar un fichero (exec(..)), abrir, leer, escribir, cerrar un fichero (open(..), read(...), write(...), close(...)).*

9. ¿Cuál es la diferencia fundamental entre un algoritmo de planificación con expulsión (preemptive) y otro que no lo sea?
En un algoritmo con expulsión, el proceso en ejecución puede ser devuelto al estado de preparado (por no haber finalizado su ráfaga de CPU) para ceder el procesador a otro proceso que se encuentra en el estado de preparado con una prioridad mayor.
10. Decidir si las siguientes afirmaciones sobre el sistema UNIX son verdaderas o falsas. No se requiere justificar las respuestas.
- UNIX es un sistema operativo de tiempo compartido.
VERDADERO. UNIX es un sistema multiprogramado e interactivo. Por tanto, es de tiempo compartido.
 - Un proceso UNIX puede crear otros procesos hijos, quienes heredarán parte de sus recursos.
VERDADERO. Los procesos hijos son inicialmente clones del padre, con lo cual heredan la mayor parte de sus recursos.
 - En un sistema UNIX sólo podremos tener un intérprete de comandos.
FALSO. El intérprete de comandos es un programa del sistema. Por tanto, no forma parte del sistema operativo, sino que únicamente utiliza las llamadas que éste proporciona. De este modo, cualquier usuario puede escribir su propio intérprete de comandos.
11. Dado un sistema multiprogramado en el que se encuentra en estado preparado un proceso orientado a E/S y una gran cantidad de procesos orientados a cómputo, ordénese de mejor a peor las siguientes estrategias de planificación de la CPU en función de que consigan mejores tiempos de retorno del proceso orientado a E/S:
- a) FCFS
 - b) SJF
 - c) SRTF
- Las estrategias que mejoran el tiempo de los procesos orientados a E/S son aquellas que premian a los procesos "cortos" (los de menor ciclo de CPU). Suponiendo que se aplica un buen estimador, SRTF es la mejor estrategia, después vendría SJF y por último FCFS.*
12. ¿Puede pasar un proceso del estado **preparado** al estado **suspendido**? Razone (brevemente) la respuesta.
No, para que un proceso se suspenda necesita pasar primero al estado de ejecución y ejecutar alguna operación que pueda suspenderle (como por ejemplo P() sobre un semáforo, wait sobre una variable condición, etc...)
13. ¿Cuál es la principal ventaja de la multiprogramación en relación con el rendimiento del sistema?
La multiprogramación aumenta tanto la productividad del sistema como el nivel de ocupación tanto del procesador como de la periferia. Esto es así puesto que permite que mientras un proceso abandona el procesador (por ejemplo, para esperar una transferencia a disco) el sistema puede seleccionar otro proceso para que ocupe el procesador.
14. Describir la función de un planificador a largo plazo, con qué colas trabaja y cuándo debe utilizarse.
El planificador a largo plazo debe cargar en la memoria los procesos de nueva creación. Utiliza la cola de trabajos para obtener qué procesos fueron creados y todavía no cargados y deposita estos en la cola de procesos preparados. La carga se efectúa cuando el sistema tiene suficiente memoria (y otros recursos) para poder albergar los nuevos procesos. Por ejemplo cuando un proceso finaliza.
15. ¿Qué diferencia fundamental existe entre un algoritmo SJF (primero el trabajo más breve, shortest job first) y el algoritmo SRTF (primero el de menor tiempo restante, shortest remaining time first)?
La diferencia fundamental es que el SJF no es expulsivo y el SRTF sí lo es.

16. Un sistema tiene los siguientes recursos: una CPU, dos discos (DISC1, DISC2) y una impresora (IMP). Existen dos tipos de trabajos según las necesidades de utilización de los recursos:

Trabajo tipo 1

CPU	DISC1	DISC1	DISC1	CPU	CPU	IMP	IMP	IMP	IMP	IMP	IMP	CPU	
0	1			4		6						12	13

Trabajo tipo 2

CPU	CPU	CPU	CPU	CPU	CPU	DISC1	CPU	CPU	CPU	DISC2	DISC2	CPU	IMP	CPU	CPU
0						6	7			10		12	13	14	16

Suponiendo que en el sistema hay un trabajo de cada tipo, calcular la utilización de la CPU, el tiempo medio de retorno en el sistema y el tiempo de espera para cada trabajo con los siguientes algoritmos de planificación de la CPU:

- FCFS
- Prioridad expulsiva (mayor prioridad al proceso de tipo 1).
- Round-Robin con $q=1$.

Suponga que el resto de las colas se gestionan con un algoritmo FCFS y que el primer proceso en el sistema es de tipo 1.

Se considera en la solución que el trabajo de tipo 1 está representado por P1, mientras que el trabajo tipo 2 se encuentra representado por P2.

a) Algoritmo de planificación FCFS.

t	Preparado	CPU	Cola DISC1	DISC1	Cola DISC2	DISC2	Cola IMP	IMP	Comentarios
0	P2	P1							Llegan P1,P2
1		P2		P1					
2		P2		P1					
3		P2		P1					
4	P1	P2							
5	P1	P2							
6	P1	P2							
7		P1		P2					
8	P2	P1							
9		P2						P1	
10		P2						P1	
11		P2						P1	
12						P2		P1	
13						P2		P1	
14		P2						P1	
15		P1						P2	
16		P2							Fin P1
17		P2							
18									Fin P2

$$Utilización_CPU = \frac{tiempo_recurso_ocupado}{Tiempo_total} = \frac{16}{18} = 0.889 \Rightarrow 89\%$$

$$Tiempo_medio_retorno = \frac{tiempo_retorno\ P1 + tiempo_retorno\ P2}{2} = \frac{16+18}{2} = 17$$

$$Tiempo_espera(P1) = 3; \quad Tiempo_espera(P2) = 2$$

b) Algoritmo de planificación con prioridad expulsiva (mayor prioridad para P1).

t	Preparado	CPU	Cola DISC1	DISC1	Cola DISC2	DISC2	Cola IMP	IMP	Comentarios
0	P2	P1							Llegan P1,P2
1		P2		P1					
2		P2		P1					
3		P2		P1					
4	P2	P1							
5	P2	P1							
6		P2						P1	
7		P2						P1	
8		P2						P1	
9				P2				P1	
10		P2						P1	
11		P2						P1	
12	P2	P1							
13		P2							Fin P1
14						P2			
15						P2			
16		P2							
17								P2	
18		P2							
19		P2							
20									Fin P2

$$Utilización_CPU = \frac{tiempo_recurso_ocupado}{Tiempo_total} = \frac{16}{20} = 0.8 \Rightarrow 80\%$$

$$Tiempo_medio_retorno = \frac{tiempo_retorno\ P1 + tiempo_retorno\ P2}{2} = \frac{13+20}{2} = 16.5$$

$$Tiempo_espera(P1) = 0; \quad Tiempo_espera(P2) = 4$$

c) Algoritmo de planificación Round-Robin con $q=1$.

t	Preparado	CPU	Cola DISC1	DISC1	Cola DISC2	DISC2	Cola IMP	IMP	Comentarios
0	P2	P1							Llegan P1,P2
1		P2		P1					
2		P2		P1					
3		P2		P1					
4	P2	P1							
5	P1	P2							
6	P2	P1							
7		P2						P1	
8		P2						P1	
9				P2				P1	
10		P2						P1	
11		P2						P1	
12		P2						P1	
13		P1				P2			
14						P2			Fin P1
15		P2							
16								P2	
17		P2							
18		P2							
19									Fin P2
20									

$$Utilización_CPU = \frac{tiempo_recurso_ocupado}{Tiempo_total} = \frac{16}{19} = 0.84 \Rightarrow 84\%$$

$$Tiempo_medio_retorno = \frac{tiempo_retorno\ P1 + tiempo_retorno\ P2}{2} = \frac{14 + 19}{2} = 16.5$$

$$Tiempo_espera(P1) = 1; \quad Tiempo_espera(P2) = 3$$

17. Dada la siguiente situación en un sistema con planificación por prioridades expulsivas:

Proceso	Instante de llegada	Tiempo de CPU	Prioridad
A	0	8	5
B	3	4	7
C	6	2	9
D	10	3	8
E	15	6	1
F	24	4	5

Suponiendo que las prioridades son crecientes con su valor, obtener el diagrama de ocupación de la CPU, el tiempo medio de retorno y el de espera de los procesos.

Solución:

t	Preparado	CPU	Comentarios
0		A	Llega A
1		A	

2		A	
3	A	B	Llega B
4	A	B	
5	A	B	
6	B,A	C	Llega C
7	B,A	C	
8	A	B	Fin C
9		A	Fin B
10	A	D	Llega D
11	A	D	
12	A	D	
13		A	Fin D
14		A	
15	E	A	Llega E
16	E	A	
17		E	Fin A
18		E	
19		E	
20		E	
21		E	
22		E	
23			Fin E
24		F	Llega F
25		F	
26		F	
27		F	
28			Fin F

$$Tiempo_medio_retorno = \frac{17 + 6 + 2 + 3 + 8 + 4}{6} = 6.67$$

$$Tiempo_medio_espera = \frac{9 + 2 + 0 + 0 + 2 + 0}{6} = 2.167$$

- 18.** El algoritmo de planificación es una versión modificada del Round-Robin tradicional que da mejor servicio a los procesos que ya se han ejecutado durante un cierto periodo de tiempo que a los recién llegados. La cola de preparados se divide en dos: una de procesos NUEVOS y otra de ACEPTADOS. Se escoge siempre para ejecución un proceso de la cola de ACEPTADOS mediante una estrategia Round-Robin, y los procesos que llegan al sistema esperan en la cola de NUEVOS hasta que pueden pasar a la de ACEPTADOS.

Cuando un proceso llega al sistema su prioridad es 0, y en cada unidad de tiempo el algoritmo calcula las prioridades para todos los procesos de la forma siguiente:

- Si un proceso está en la cola de NUEVOS, se incrementa su prioridad en un factor a.
- Si un proceso está en la cola de ACEPTADOS se incrementa su prioridad en un factor b.
- Cuando la prioridad de un proceso NUEVO se hace mayor o igual a la de cualquier proceso de la cola de ACEPTADOS, el proceso NUEVO se inserta en ella. En caso de que se vacíe la cola de ACEPTADOS, se introduce en ella el proceso más prioritario de la cola de NUEVOS. Un proceso es más prioritario cuanto mayor sea el valor numérico de su prioridad.

- a) Suponiendo que a=2, b=1 y q=1, y la siguiente situación:

Proceso	Instante de llegada	Tiempo de CPU
A	0	5
B	1	4
C	3	2
D	9	6
E	11	3

Represente el diagrama de ocupación de la CPU y calcule los tiempos medios de espera y de retorno en el sistema.

- b) Analice el comportamiento del algoritmo en los siguientes casos: $a > b$, $b \geq a$, siendo $a, b > 0$.

Solución:

a) Para obtener la solución que se presenta en la tabla siguiente se ha tenido en cuenta los siguientes criterios:

En cada instante de tiempo en primer lugar se calculan las prioridades y a continuación actúa el planificador.

A la cola de ACEPTADOS llegan en primer lugar los procesos que vienen de la cola de NUEVOS, y a continuación el que abandona la CPU.

PREPARADOS					PRIORIDADES				
t	nuevos	aceptados	CPU	Comentarios	A	B	C	D	E
0			A	Llega A*	0				
1	B		A	Llega B	1	0			
2		A	B	B aceptado	2	2			
3	C	B	A	Llega C	3	3	0		
4	C	A	B		4	4	2		
5	C	B	A		5	5	4		
6		A, C	B	C aceptado	6	6	6		
7		B, A	C		7	7	7		
8		C, B	A		8	8	8		
9	D	C	B	Llega D, Fin A	-	9	9	0	
10	D		C	Fin B		-	10	2	
11	E		D	Llega E, Fin C			-	4	0
12	E		D					5	2
13	E		D					6	4
14	E		D					7	6
15		D	E	E aceptado				8	8
16		E	D					9	9
17		D	E					10	10
18		E	D					11	11
19			E	Fin D				-	12
20				Fin E					-

* Se considera que llega A y como no hay ningún proceso en la cola de ACEPTADOS, el proceso A pasa directamente de la cola de NUEVOS a la de ACEPTADOS.

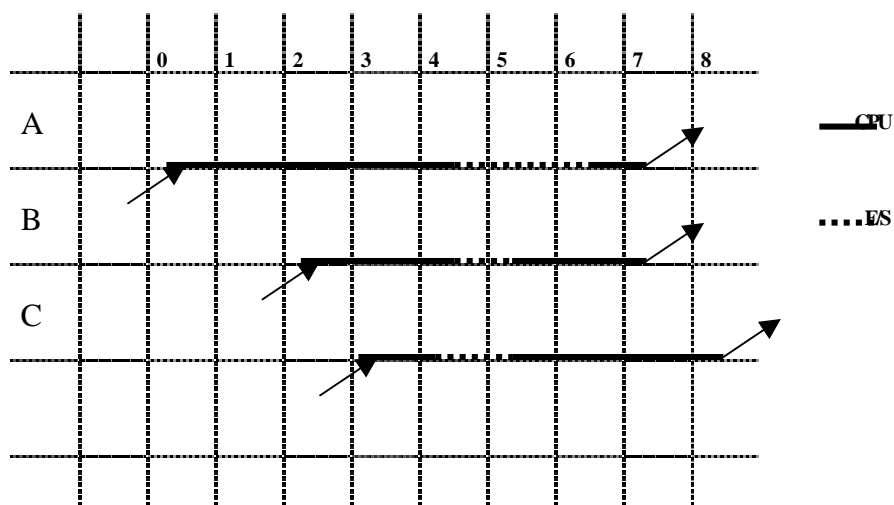
$$Tiempo_medio_retorno = \frac{9+9+8+10+9}{5} = 9$$

$$Tiempo_medio_espera = \frac{4+5+6+4+6}{5} = 5$$

- b) Si $a \gg b$ los nuevos procesos pasarían a la cola de ACEPTADOS directamente y el algoritmo de planificación se correspondería con el Round-Robin tradicional. Si $b \geq a$ los procesos nuevos no tendrían posibilidad de ser aceptados mientras hubiese procesos en la cola de ACEPTADOS y el algoritmo de planificación sería equivalente a FCFS.

19. Sea un algoritmo de planificación multicolos con realimentación donde la primera cola (cola 0) se gestiona con un Round-Robin de quantum $q=1$, y la segunda cola (cola 1) se gestiona con un algoritmo SRTF. La planificación entre colas es del tipo prioridades expulsiva, la cola más prioritaria es la 0. Un proceso pasará de la cola 0 a la 1 cuando se agote su quantum q sin finalizar su ejecución. Los procesos nuevos, y los procedentes del estado SUSPENDIDO, entran por la cola 0.

Suponiendo que el s.o. no consume tiempo y que las operaciones de E/S se efectúan sobre el mismo dispositivo (gestionado de forma FCFS), dibujar el diagrama temporal de ocupación del procesador y del dispositivo de E/S para los procesos A, B y C que se ilustran a continuación.



En la figura anterior, las líneas continuas representan tiempo de procesamiento y las discontinuas muestran tiempo dedicado a operaciones de E/S. El gráfico refleja el comportamiento y duración de las operaciones de cada proceso en caso de que el sistema fuese monoprogramado.

Solución:

t	PREPARADOS			SUSPENDIDOS		Comentarios
	Cola 0	Cola 1	CPU	Cola E/S	E/S	
0			A			Llega A
1			A			A en Cola 1 a CPU
2		A	B			Llega B
3		B,A	C			Llega C
4		A	B		C	
5		C, A	C		B	C en Cola 0 a CPU

6		C, A	B			B en Cola 0 a CPU
7		C, A	B			B en Cola 1 a CPU
8		C	A			Fin B
9		C	A			
10			C		A	
11			C		A	
12			A			Fin C
13						Fin A
14						

20. Suponga que a un sistema llegan de forma simultánea 5 procesos (A, B, C, D, E), siendo el perfil de ejecución de cada uno de ellos el siguiente: 1000 unidades de tiempo de CPU, 5 de IMPRESORA y otras 1000 de CPU. Si en dicho sistema se aplica una política de planificación circular (Round-Robin) con quantum $q=5$, ¿Cuáles serán los tiempos de retorno y espera para cada uno de los 5 procesos?. Justifique la respuesta. (Sugerencia: no es muy recomendable hacer una traza completa).

Solución:

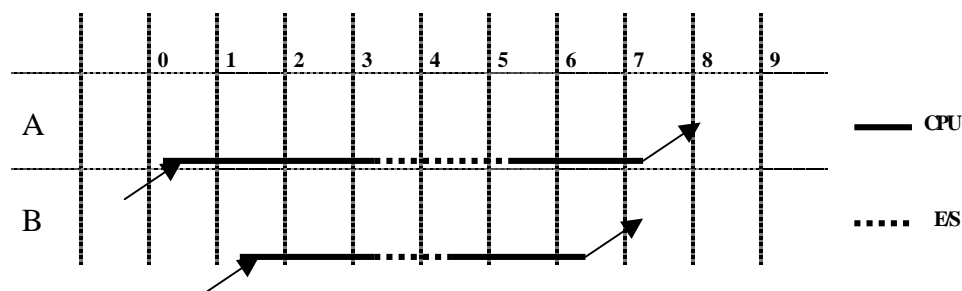
Al ser una planificación Round-Robin, los procesos van intercalándose cada 5 unidades de tiempo en la CPU, creando secuencias del tipo AAAAA, BBBB, CCCCC, DDDDD, EEEEE, AAAAA, BBBB, CCCCC, DDDDD, EEEEE..... Al coincidir el tiempo de ráfaga de impresora (5 unidades de tiempo) con el quantum ($q=5$), no se producen retardos en el acceso a la impresora (los procesos no esperan en la cola de la impresora), y por tanto, el tiempo de retorno sólo depende del consumo de CPU.

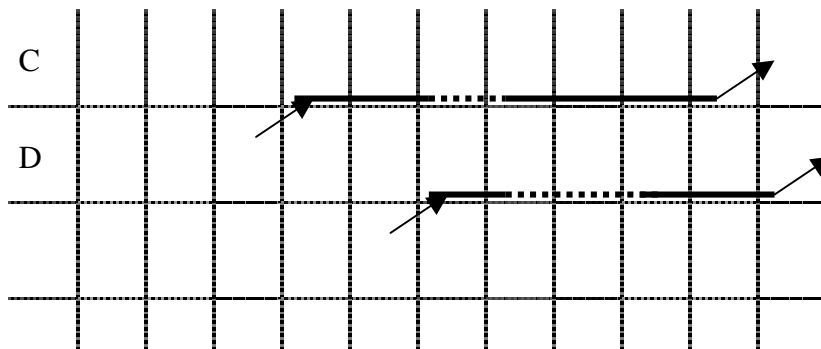
Cada proceso consume 2000 unidades de tiempo de CPU, como hay 5 procesos, se requerirán 10000 unidades para que todos terminen, aunque hay que tener en cuenta que no todos terminan en el mismo instante. El proceso A es el primero que finaliza, y lo hace cuando a los cuatro restantes procesos (B, C, D, E) les falta todavía por consumir un quantum de tiempo ($4 \times 5 = 20$), a continuación finalizará el proceso B y lo hace cuando a los tres restantes procesos (C, D, E) les falta todavía por consumir un quantum de tiempo ($3 \times 5 = 15$) y así sucesivamente hasta que finalicen los cinco procesos. Por tanto, el tiempo de retorno para el proceso A será de 9980, para B de 9985, para C de 9990, para D de 9995 y para E de 10000.

Dado que el tiempo de espera de cada proceso es igual al de retorno menos el tiempo de actividad ($2000 \text{ CPU} + 5 \text{ IMPRESORA} = 2005$), el tiempo de espera para el proceso A será de 7975, para el B de 7980, para el C de 7985, para el D de 7990 y para el E de 7995.

21. Sea un algoritmo de planificación multicolos con realimentación donde cada cola se gestiona con un Round-Robin de quantum $q_i = 2^i$. La planificación entre colas es de tipo prioridades expulsivas, la cola más prioritaria es la 0. Un proceso pasará de la cola i a la cola $i+1$ cuando agote su quantum q_i sin finalizar su ejecución. Los procesos nuevos y los procedentes del estado SUSPENDIDO entran por la cola 0.

Suponiendo que el s.o. no consume tiempo y que las operaciones de E/S se efectúan sobre el mismo dispositivo (gestionado de forma FCFS), dibujar el diagrama temporal de ocupación del procesador y del dispositivo de E/S para los procesos A, B, C y D que se ilustran a continuación. Muéstrese también la situación de la cola de PREPARADOS.





En la figura anterior, las líneas continuas representan tiempo de procesamiento y las discontinuas muestran tiempo dedicado a operaciones de E/S. El gráfico refleja el comportamiento y duración de las operaciones de cada proceso en caso de que el sistema fuese monoprogramado.

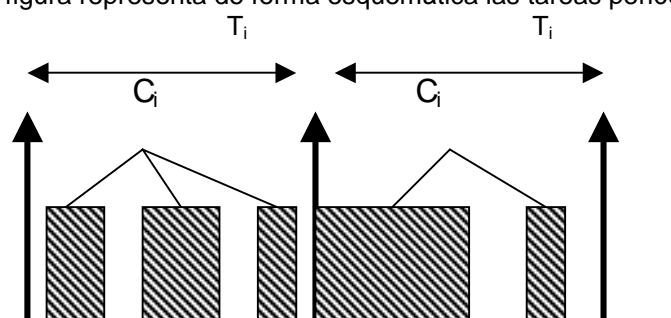
Solución:

Las colas se gestionan con un algoritmo de planificación RR con $q=2^i$, con lo cual la cola 0 tendrá un quantum $q_0=2^0=1$, la cola 1 $q_1=2^1=2$, la cola 2 $q_2=2^2=4, \dots$

Los procesos marcados en la tabla con * se han quedado a medio ejecutar su quantum de tiempo $q_1=2$ correspondiente a la cola 1. El criterio seguido en este caso es el de ponerlo en cabeza de procesos de la cola 1, la siguiente vez que actúe el planificador y seleccione un proceso de dicha cola, lo seleccionará en primer lugar y lo ejecutará durante un único intervalo de tiempo.

t	PREPARADOS		CPU	SUSPENDIDOS		Comentarios
	Cola 0	Cola 1		Cola E/S	E/S	
0			A			Llega A
1		A	B			Llega B
2		B, A	C			Llega C
3		C, B	A			
4		C, B, A*	D			Llega D, Expulsa A
5		C, B	A		D	
6		C	B	A	D	
7		C	D	B	A	D en cola 0 a CPU
8		D	C	B	A	
9		D	A	C	B	A en cola 0 a CPU
10		A, D	B		C	B en cola 0 a CPU
11		B, A, D	C			C en cola 0 a CPU
12		C, B, A	D			
13		C, B	A			Fin D
14		C	B			Fin A
15			C			Fin B
16			C			
17						Fin C

22. Un sistema de control de una nave espacial está formado por tres tareas periódicas P_1, P_2, P_3 . Una tarea periódica P_i es una tarea que repite sus cálculos cada intervalo de tiempo T_i , denominado periodo y se caracteriza por los siguientes parámetros: duración del periodo T_i y duración de los cálculos C_i (tiempo de CPU) que realiza en cada periodo ($C_i \leq T_i$). La siguiente figura representa de forma esquemática las tareas periódicas.

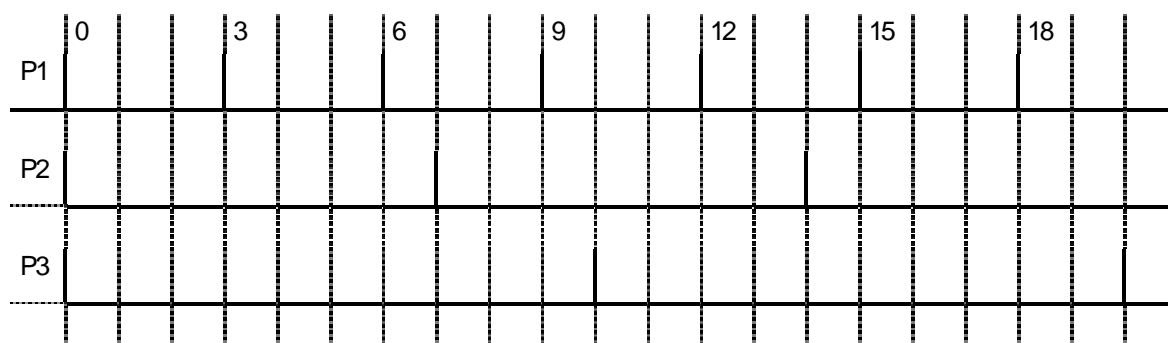


Una tarea periódica se activa al principio de su periodo y se suspende desde que termina sus cálculos, hasta el principio del siguiente periodo. La tarea debe terminar sus cálculos de un periodo antes de que comience el siguiente periodo. En caso contrario, el sistema se dice que falla y se pasa a una situación de emergencia.

Para evitar los fallos, el sistema utiliza un algoritmo de planificación denominado “*prioridad a la tarea más urgente*”, que es un algoritmo de prioridades con expulsión en el que la prioridad de una tarea es su urgencia, la cual se define a continuación: La *urgencia* de una tarea en el instante t es el tiempo que falta desde t hasta el comienzo de su siguiente periodo. Se pide:

- a) Represente el diagrama de ocupación de la CPU del instante 0 al instante 20 (ambos inclusive) para la carga expuesta en la siguiente tabla. Suponga que los periodos de todas las tareas comienzan en el instante 0.

Tarea	Periodo T_i	Tiempo de CPU
P1	3	1
P2	7	3
P3	10	2



- b) Dadas 3 tareas con periodos iguales, ¿Cuál es la condición para que no ocurra una situación de emergencia?.

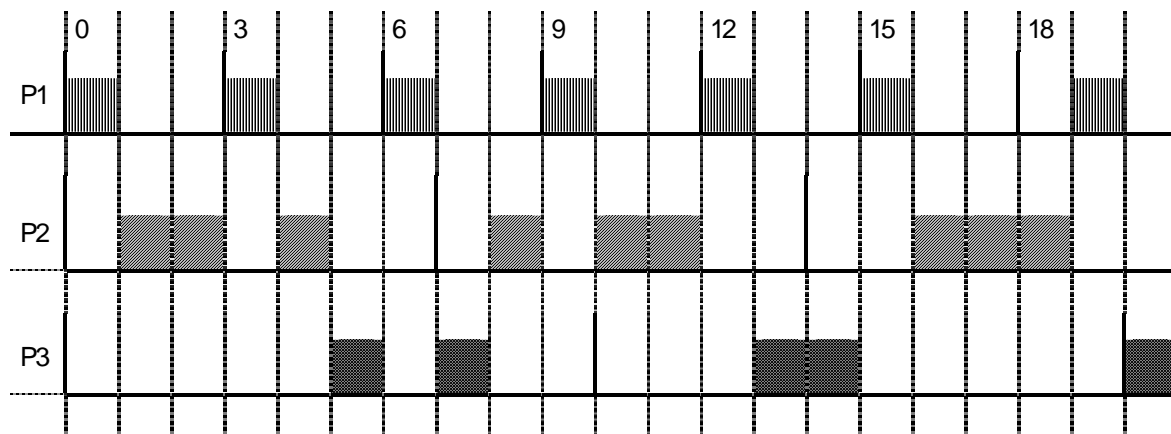
Solución

a) En cada instante de tiempo se ha representado en la tabla la prioridad de cada proceso, entre paréntesis, el proceso más prioritario es aquél que menos tiempo le falta para finalizar su periodo y por tanto, con menor número de prioridad. Los procesos se encontrarán en el estado de preparado al inicio de su periodo y pasarán a suspendidos una vez hayan cumplido su tiempo de CPU, hasta el inicio de su periodo.

t	Preparados	CPU	Suspendidos	Comentarios
0	P2(7), P3(10)	P1(3)		Inicio periodo P1(3), P2(7), P3(10)
1	P3(9)	P2(6)	P1	Fin P1
2	P3(8)	P2(5)	P1	
3	P3(7), P2(4)	P1(3)		Inicio periodo P1(3)
4	P3(6)	P2(3)	P1	Fin P1
5		P3(5)	P1, P2	Fin P2
6	P3(4)	P1(3)	P2	Inicio periodo P1(3)
7	P2(7)	P3(3)	P1	Inicio periodo P2(7), Fin P1
8		P2(6)	P1, P3	Fin P3
9	P2(5)	P1(3)	P3	Inicio periodo P1(3)
10	P3(10)	P2(4)	P1	Inicio periodo P3(10), Fin P1

11	P3(9)	P2(3)	P1	
12	P3(8)	P1(3)	P2	Inicio periodo P1(3), Fin P2
13		P3(7)	P2, P1	Fin P1
14	P2(7)	P3(6)	P1	Inicio periodo P2(7)
15	P2(6)	P1(3)	P3	Inicio periodo P1(3), Fin P3
16		P2(5)	P3, P1	Fin P1
17		P2(4)	P3, P1	
18	P1(3)	P2(3)	P3	Inicio periodo P1(3)
19		P1(2)	P3, P2	Fin P2
20		P3(10)	P2, P1	Inicio periodo P3(10), Fin P1
21				

Si se representan estos resultados en un cronograma por proceso se obtendría lo siguiente:



b) La condición necesaria para que no ocurra una condición de emergencia es que la suma de los tiempos de cómputo de las tareas no supere el periodo.

$$\sum_{i=1}^3 C_i \leq T_i$$

23. Dados los siguientes procesos a ejecutar en el sistema:

Proceso	Instante de llegada	Tiempo de CPU
A	0	3
B	1	5
C	3	2
D	9	5
E	11	5

Representar el diagrama de ocupación de la CPU y calcular los tiempos medios de espera y de retorno en el sistema para las siguientes estrategias de planificación:

- Round-Robin con $q=2$
- Colas multinivel con realimentación, donde cada cola se gestiona con RR de quantum $q_i = 2^i \cdot q_0$, siendo $q_0=1$, la llegada de un proceso a una cola más prioritaria nunca interrumpe la ejecución del proceso (planificación entre colas del tipo prioridades no expulsivas) y la cola más prioritaria es la cola 0. Un proceso i pasa a pertenecer a una cola $i+1$ cuando agota su quantum q_i sin finalizar su ejecución. La cola por la que entran los procesos nuevos es la cola 0.

Solución

a) Round- Robin con $q=2$.

t	Preparados	CPU	Comentarios
0		A	Llega A
1	B	A	Llega B
2	A	B	
3	C, A	B	Llega C
4	B, C	A	
5	B	C	Fin A
6	B	C	
7		B	Fin C
8		B	
9	B	D	Llega D
10	B	D	
11	D	B	
12	E	D	Llega E, Fin B
13	E	D	
14	D	E	
15	D	E	
16	E	D	
17		E	Fin D
18		E	
19		E	
20			Fin E

$$Tiempo_medio_retorno = \frac{5+11+4+8+8}{5} = 7.2$$

$$Tiempo_medio_espera = \frac{2+5+3+3+3}{5} = 3.2$$

b) **Colas multinivel con realimentación, cada cola RR con $q_i=2i*q_0$. Teniendo en cuenta que $q_0=1$, la cola 1 tendrá un quantum $q_1=2$, la cola 2 un $q_2=4$,....**

PREPARADOS				CPU	Comentarios
t	Cola 0	Cola 1	Cola 2		
0				A	Llega A en cola 0
1		A		B	Llega B en cola 0
2		B		A	
3	C	B		A	Llega C en cola 0
4		B		C	Fin A
5		C		B	
6		C		B	
7			B	C	
8				B	Fin C
9	D			B	Llega D en cola 0
10				D	Fin B
11				D	Llega D en cola 1
12	E			D	Llega E en cola 0
13			D	E	
14			D	E	Llega E en cola 1
15			D	E	
16			E	D	
17			E	D	
18				E	Fin D

19				E	
20					Fin E

$$Tiempo_medio_retorno = \frac{4+9+5+9+8}{5} = 7$$

$$Tiempo_medio_espera = \frac{1+4+3+4+3}{5} = 3$$

- 24.** Un sistema dispone de un planificador a largo plazo (PLP) y de otro a corto plazo (PCP), que funcionan de la siguiente manera: el PCP utiliza un algoritmo con prioridades expulsivas y el PLP utiliza una estrategia FCFS. Los procesos nuevos entran por el PLP. Se admiten en memoria un máximo de tres procesos. El PLP pasa un proceso al PCP cuando hay menos de tres procesos ya admitidos. Partiendo de la siguiente tabla de procesos y teniendo en cuenta que a menor número más prioridad.

Proceso	Instante de llegada	Tiempo de CPU	Prioridad
A	0	2	4
B	1	4	3
C	3	4	2
D	5	1	1
E	6	2	3

Representa el diagrama de ocupación de la CPU y calcula los tiempos medios de espera y de retorno en el sistema.

Solución

El PLP será el encargado de seleccionar de la cola de pendientes para pasarlos a la cola de preparados, con el criterio de FCFS y teniendo en cuenta que el total entre procesos preparados más el de ejecución no puede superar las 3 unidades. El PCP es el encargado de decidir a cuál de los procesos de la cola de preparados se le adjudica la CPU, con el criterio de prioridades expulsivas. En la tabla se indica entre paréntesis el número de prioridad de cada proceso.

t	Pendientes	Preparados	CPU	Comentarios
0			A(4)	Llega A(4)
1		A(4)	B(3)	Llega B(3), PLP pasa B a preparado
2		A(4)	B(3)	
3		B(3), A(4)	C(2)	Llega C(2), PLP pasa C a preparado
4		B(3), A(4)	C(2)	
5	D(1)	B(3), A(4)	C(2)	Llega D(1), en memoria hay 3 procesos
6	E(3), D(1)	B(3), A(4)	C(2)	Llega E(3)
7	E(3)	B(3), A(4)	D(1)	Fin C, PLP pasa D a preparado
8		E(3), A(4)	B(3)	Fin D, PLP pasa E a preparado
9		E(3), A(4)	B(3)	PCP asigna la CPU a B
10		A(4)	E(3)	Fin B, PCP asigna la CPU a E
11		A(4)	E(3)	
12			A(4)	Fin E
13				Fin A

$$Tiempo_medio_retorno = \frac{13+9+4+3+6}{5} = 7$$

$$Tiempo_medio_espera = \frac{11+5+0+2+4}{5} = 4.4$$

25. Sean los siguientes procesos a ejecutar en un sistema:

Proceso	Instante de llegada	Orden de llegada	Tiempo de CPU	Prioridad
A	0	1	8	3
B	0	2	1	1
C	0	3	2	3
D	0	4	1	4
E	0	5	5	2

Representa el diagrama de ocupación de la CPU y calcula los tiempos medios de espera y de retorno en el sistema para los siguientes algoritmos de planificación:

- FCFS
- Round-Robin con $q=1$.
- SJF
- Prioridades no expulsivas, considere mayor número de prioridad más prioridad.

Solución

a) FCFS.

t	Preparados	CPU	Comentarios
0	E, D, C, B	A	Llegan A, B, C, D, E
1	E, D, C, B	A	
2	E, D, C, B	A	
3	E, D, C, B	A	
4	E, D, C, B	A	
5	E, D, C, B	A	
6	E, D, C, B	A	
7	E, D, C, B	A	
8	E, D, C	B	Fin A
9	E, D	C	Fin B
10	E, D	C	
11	E	D	Fin C
12		E	Fin D
13		E	
14		E	
15		E	
16		E	
17			Fin E

b) RR con $q=1$.

t	Preparados	CPU	Comentarios
0	E, D, C, B	A	Llegan A, B, C, D, E
1	A, E, D, C	B	
2	A, E, D	C	Fin B
3	C, A, E	D	
4	C, A	E	Fin D
5	E, C	A	
6	A, E	C	
7	A	E	Fin C
8	E	A	
9	A	E	
10	E	A	
11	A	E	
12	E	A	
13	A	E	
14		A	Fin E
15		A	
16		A	
17			Fin A

c) SJF.

t	Preparados	CPU	Comentarios
0	E, D, C, A	B	Llegan A, B, C, D, E
1	E, C, A	D	Fin B
2	E, A	C	Fin D
3	E, A	C	
4	A	E	Fin C
5	A	E	
6	A	E	
7	A	E	
8	A	E	
9		A	Fin E
10		A	
11		A	
12		A	
13		A	
14		A	
15		A	
16		A	
17			Fin A

d) Prioridades no expulsivas. Entre paréntesis se indica la prioridad de los procesos.

t	Preparados	CPU	Comentarios
0	E(2), C(3), B(1), A(3)	D(4)	Llegan A(3), B(1), C(3), D(4), E(2)
1	E(2), C(3), B(1)	A(3)	Fin D
2	E(2), C(3), B(1)	A(3)	
3	E(2), C(3), B(1)	A(3)	
4	E(2), C(3), B(1)	A(3)	
5	E(2), C(3), B(1)	A(3)	
6	E(2), C(3), B(1)	A(3)	
7	E(2), C(3), B(1)	A(3)	
8	E(2), C(3), B(1)	A(3)	
9	E(2), B(1)	C(3)	Fin A

10	E(2), B(1)	C(3)	
11	B(1)	E(2)	Fin C
12	B(1)	E(2)	
13	B(1)	E(2)	
14	B(1)	E(2)	
15	B(1)	E(2)	
16		B(1)	Fin E
17			Fin B

Los tiempos medios de retorno y espera de estos algoritmos de planificación se resumen en la tabla siguiente:

	FCFS	RR	SJF	Prioridades
Tiempo medio retorno	13	8.8	6.6	10.8
Tiempo medio espera	9.6	5.4	3.2	7.6

26. Suponga que los siguientes procesos llegan en el instante indicado:

Proceso	Instante de llegada	Tiempo de CPU
A	0.0	8
B	0.4	4
C	1	1

Calcule el tiempo medio de retorno para estos procesos con los siguientes algoritmos de planificación:

- FCFS.
- SJF.
- Se supone que SJF se utiliza para mejorar las prestaciones, pero obsérvese que en el apartado anterior se selecciona el trabajo A en el instante 0 porque se desconoce que poco después van a llegar dos trabajos con menor ráfaga de CPU. Calcúlese el tiempo medio de retorno si la CPU se deja desocupada durante la primera unidad de tiempo y luego se aplica SJF. Obsérvese que los procesos A y B esperarán durante este tiempo. A este algoritmo se le podría denominar “*planificación con conocimiento de futuro*”.

Solución

a) *FCFS*.

t	Preparados	CPU	Comentarios
0	B	A	Llegan A, B
1	C, B	A	Llega C
2	C, B	A	
3	C, B	A	
4	C, B	A	
5	C, B	A	
6	C, B	A	
7	C, B	A	
8	C	B	Fin A
9	C	B	
10	C	B	
11	C	B	
12		C	Fin B
13			Fin C

b) *SJF*

t	Preparados	CPU	Comentarios
0	B	A	Llegan A, B
1	C, B	A	Llega C
2	C, B	A	
3	C, B	A	
4	C, B	A	
5	C, B	A	
6	C, B	A	
7	C, B	A	
8	B	C	Fin A
9		B	Fin C
10		B	
11		B	
12		B	
13			Fin B

c) SJF con CPU desocupada durante el primer intervalo de tiempo.

t	Preparados	CPU	Comentarios
0	A, B		Llegan A, B
1	B, A	C	Llega C
2	A	B	Fin C
3	A	B	
4	A	B	
5	A	B	
6		A	Fin B
7		A	
8		A	
9		A	
10		A	
11		A	
12		A	
13		A	
14			Fin A

Los tiempos medios de retorno de estos algoritmos de planificación se resumen en la tabla siguiente, fíjese que a pesar de que en la propuesta c se tarda un intervalo más de tiempo para ejecutar los tres procesos, el tiempo medio de retorno para dicha propuesta es menor.

	FCFS	SJF	SJF con Conocimiento de Futuro
Tiempo medio retorno	11	10	7.3

27. En un sistema multiprogramado se tienen dos procesos, A y B, con las siguientes distribución de su tiempo de ejecución.

Proceso A: 3 CPU + 4 DISCO + 6 CPU + 4 IMPRESORA + 3 CPU

Proceso B: 2 CPU + 5 DISCO + 3 CPU + 3 IMPRESORA + 2 CPU

El proceso A llega al sistema en el instante 0, mientras que el B lo hace en el instante 2. Sólo hay un disco y una impresora. Los periféricos utilizan un algoritmo de planificación FCFS. Dibuje el diagrama de ocupación de la CPU y dé el tiempo medio de retorno para cada uno de los siguientes supuestos:

- Planificación del procesador Round-Robin con $q=1$.
- Planificación del procesador prioridades expulsivo, siendo mayor la de B.

Solución

a) *RR con q=1*

t	Preparado	CPU	Cola DISCO	DISCO	Cola IMP	IMP	Comentarios
0		A					Llega A
1		A					
2	A	B					Llega B
3	B	A					
4		B		A			
5			B	A			
6			B	A			
7			B	A			
8		A		B			
9		A		B			
10		A		B			
11		A		B			
12		A		B			
13	A	B					
14	B	A					
15		B				A	
16		B				A	
17					B	A	
18					B	A	
19		A				B	
20		A				B	
21		A				B	
22		B					Fin A
23		B					
24							Fin B

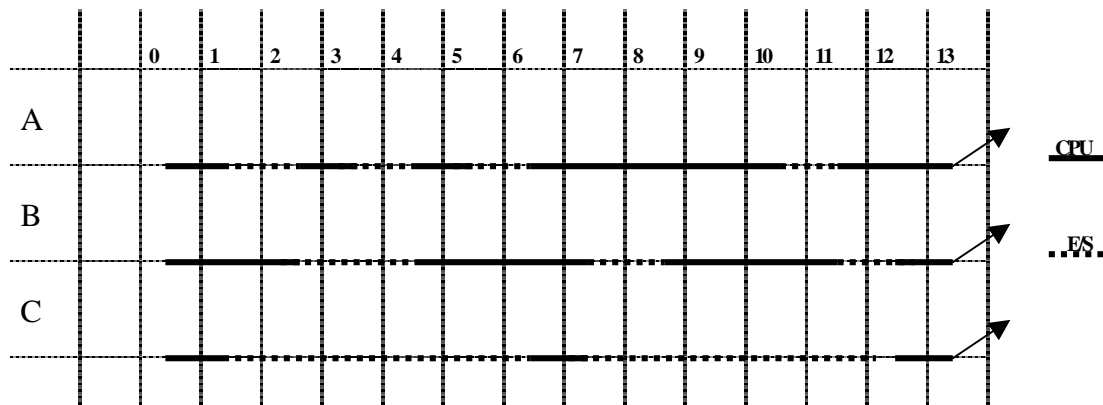
b) *Prioridades expulsivas siendo más prioritario B.*

t	Preparado	CPU	Cola DISCO	DISCO	Cola IMP	IMP	Comentarios
0		A					Llega A
1		A					
2	A	B					Llega B
3	A	B					
4		A		B			
5			A	B			
6			A	B			
7			A	B			
8			A	B			
9		B		A			
10		B		A			
11		B		A			
12				A		B	
13		A				B	
14		A				B	
15	A	B					
16	A	B					
17		A					Fin B
18		A					
19		A					
20		A					
21						A	
22						A	
23						A	

24						A	
25		A					
26		A					
27		A					
28							Fin A

	RR	Prioridad expulsiva
Tiempo medio espera	4.5	4
Tiempo medio retorno	22	21.5

28. Supóngase un sistema multiprogramado con un planificador expulsivo basado en prioridades. Se tienen 3 procesos A, B y C. La prioridad de A es mayor que la de B, y ésta mayor que la de C. Sólo hay un único dispositivo de E/S que se gestiona con FCFS. Represente mediante un diagrama temporal la ocupación de la CPU, del periférico de E/S y del estado preparado, teniendo en cuenta el siguiente esquema para los procesos A, B y C:



Solución

t	Preparado	CPU	Cola E/S	E/S	Comentarios
0	C, B	A			Llega A, B y C
1	C	B		A	
2	B, C	A			
3	C	B		A	
4	C	A		B	
5		C	A	B	
6		B	C	A	
7	B	A		C	
8	B	A		C	
9	B	A		C	
10	B	A		C	
11		B	A	C	
12	C	B		A	
13	C	A		B	
14	B, C	A			
15	C	B			Fin A
16	C	B			
17	C	B			
18		C		B	

19		B		C	
20				C	Fin B
21				C	
22				C	
23				C	
24		C			
25					Fin C

29. Se tienen cuatro procesos en un sistema, P1, P2, P3, P4, de forma que cada uno de ellos llega al sistema en los instantes 0, 1, 2 y 10, respectivamente. Además, cada uno necesita una ráfaga de CPU de duración 6, 6, 5 y 8 respectivamente. Indique cual será el tiempo promedio de espera si utilizamos los siguientes algoritmos de planificación de la CPU

a) SRTF

b) Prioridades expulsivas, siendo más prioritario cuanto más tarde se llegue.

Solución

a) SRTF

t	Preparado	CPU	Comentarios
0		P1	Llega P1
1	P2	P1	Llega P2
2	P3, P2	P1	Llega P3
3	P3, P2	P1	
4	P3, P2	P1	
5	P3, P2	P1	
6	P2	P3	Fin P1
7	P2	P3	
8	P2	P3	
9	P2	P3	
10	P4, P2	P3	Llega P4
11	P4	P2	
12	P4	P2	
13	P4	P2	
14	P4	P2	
15	P4	P2	
16	P4	P2	
17		P4	Fin P2
18		P4	
19		P4	
20		P4	
21		P4	
22		P4	
23		P4	
24		P4	
25			Fin P4

$$\text{Tiempo}_{\text{medio}_{\text{espera}}} = \frac{0 + 10 + 4 + 7}{4} = 5.25$$

b) Prioridades expulsivas, siendo el más prioritario el que más tarde llega.

t	Preparado	CPU	Comentarios
0		P1	Llega P1
1	P1	P2	Llega P2
2	P2, P1	P3	Llega P3
3	P2, P1	P3	
4	P2, P1	P3	
5	P2, P1	P3	
6	P2, P1	P3	
7	P1	P2	Fin P3
8	P1	P2	
9	P1	P2	
10	P2, P1	P4	Llega P4
11	P2, P1	P4	
12	P2, P1	P4	
13	P2, P1	P4	
14	P2, P1	P4	
15	P2, P1	P4	
16	P2, P1	P4	
17	P2, P1	P4	
18		P2	Fin P4
19		P2	
20		P1	Fin P2
21		P1	
22		P1	
23		P1	
24		P1	
25			Fin P1

$$Tiempo_medio_espera = \frac{19+13+0+0}{4} = 8$$

- 30.** Se tiene un sistema operativo en que el algoritmo de planificación consta de cuatro clases de prioridad numeradas del 0 al 3. El algoritmo de planificación es Round-Robin para las clases 0, 1 y 2, y es FCFS para la clase 3. La clase más prioritaria es la 0. Los cuantos de tiempo, q_i , para las clases 0 a la 2 vienen dados por la siguiente fórmula: $q_i = i + 1$. El algoritmo de planificación inter-colas es expulsivo.

Los procesos que entran en el sistema son admitidos inicialmente en la clase más prioritaria (0), existiendo un mecanismo de degradación de la prioridad obedeciendo a la siguiente regla: “un proceso permanece en su clase hasta que ha consumido dos cuantos de tiempo, tras lo cual es degradado a la clase de prioridad inmediatamente inferior”. Todo proceso que llega a la clase 3 permanece en ella hasta que termina su ejecución.

Calcular el tiempo de terminación y la clase en que terminan tres procesos, P1, P2, P3, que llegan en el instante cero (en el orden P1, P2, P3), y cuyas duraciones respectivas son 4, 8 y 12.

Solución

t	Preparado C3	Preparado C2	Preparado C1	Preparado C0	CPU	Eventos
0				P3 P2	P1	Llegan P1, P2, P3
1				P1 P3	P2	RR(P1)
2				P2 P1	P3	RR(P2)
3				P3 P2	P1	RR(P3)
4			P1	P3	P2	RR(P1)
5			P2 P1		P3	RR(P2)
6			P3 P2		P1	RR(P3)

7			P3 P2		P1	
8			P3		P2	Fin P1
9			P3		P2	
10			P2		P3	RR(P2)
11			P2		P3	
12			P3		P2	RR(P3)
13			P3		P2	
14		P2			P3	RR(P2)
15		P2			P3	
16		P3			P2	RR(P3)
17		P3			P2	
18					P3	Fin P2
19					P3	
20					P3	
21					P3	RR(P3)
22					P3	
23					P3	
24						Fin P3

31. Se tiene un sistema operativo en el que el algoritmo de planificación tiene tres clases de prioridad numeradas de 0 a 2. Dicho algoritmo de planificación sigue las reglas siguientes:

- La política de planificación de las clases 0 y 1 es de tipo Round-Robin, con cuantos de duración 1 y 2 unidades de tiempo, respectivamente. Los procesos nuevos siempre llegan a la cola 0. Un proceso pasa de una de estas colas a la siguiente cuando ha finalizado su cuanto sin terminar el ciclo de CPU iniciado.

- La política de planificación de la clase 2 es SRTF (Shortest Remaining Time First) o "Primero el que tenga el menor tiempo restante"

- La política entre colas es expulsiva por prioridades, siendo $\text{prioridad}(0) > \text{prioridad}(1) > \text{prioridad}(2)$.

Suponga que llegan al sistema cuatro procesos, con las características siguientes:

Proceso	Instante de llegada	Tiempo de CPU
A	0	7
B	2	3
C	4	6
D	6	4

Se pide: Calcule el instante de terminación de cada proceso. ¿Qué problema existe en la política planteada desde el punto de vista de la equidad entre procesos?, ¿Cómo se podría solucionar?

Solución

<i>t</i>	<i>Q0</i>	<i>Q1</i>	<i>Q2</i>	<i>CPU</i>	<i>Eventos</i>
0				A	Llega A
1				A	RR(A)
2		A		B	Llega B
3		B		A	RR(B)
4		B	A	C	Llega C, RR(A)
5		C	A	B	RR(C)
6		C B	A	D	Llega D
7		D C	A	B	RR(D)
8		D	A	C	Fin B
9		D	A	C	
10			C A	D	RR(C)

11			A	C	B	C a Disco
12			A		C	Fin B
13			A		C	
14		A	C			C de nivel 1 a CPU
15			A		C	C a Disco
16					C	Fin A
17			C			
18						Fin C

	A	B	C
Tiempo espera	6	2	7
Tiempo retorno	16	12	18

33. Se tienen cuatro procesos en un sistema con las siguientes características:

Proceso	Instante de llegada	Tiempo de CPU
P1	0	8
P2	2	4
P3	5	9
P4	12	3

Indique cual será el tiempo promedio de espera si se utilizan los siguientes algoritmos de planificación:

- SRTF
- Prioridades expulsivas, siendo más prioritario el proceso que llega más tarde.
- Turno rotatorio (Round-Robin) con cuanto de tiempo $q=2$.

Solución

a) *SRTF*

<i>t</i>	<i>PREPARADO</i>	<i>CPU</i>	<i>Eventos</i>
0		P1	Llega P1
1		P1	
2	P1	P2	Llega P2
3	P1	P2	
4	P1	P2	
5	P3, P1	P2	Llega P3
6	P3	P1	Fin P2
7	P3	P1	
8	P3	P1	
9	P3	P1	
10	P3	P1	
11	P3	P1	
12	P3	P4	Llega P4, Fin P1
13	P3	P4	
14	P3	P4	
15		P3	Fin P4
16		P3	
17		P3	
18		P3	
19		P3	
20		P3	
21		P3	
22		P3	
23		P3	
24			Fin P3

b) *Prioridades expulsivas, más prioritario el proceso que llega más tarde.*

t	PREPARADO	CPU	Eventos
0		P1	Llega P1
1		P1	
2	P1	P2	Llega P2
3	P1	P2	
4	P1	P2	
5	P2, P1	P3	Llega P3
6	P2, P1	P3	
7	P2, P1	P3	
8	P2, P1	P3	
9	P2, P1	P3	
10	P2, P1	P3	
11	P2, P1	P3	
12	P3, P2, P1	P4	Llega P4
13	P3, P2, P1	P4	
14	P3, P2, P1	P4	
15	P2, P1	P3	Fin P4
16	P2, P1	P3	
17	P1	P2	Fin P3
18		P1	Fin P2
19		P1	
20		P1	
21		P1	
22		P1	
23		P1	
24			Fin P1

c) *Turno rotatorio con cuanto de tiempo q=2.*

t	PREPARADO	CPU	Eventos
0		P1	Llega P1
1		P1	
2	P1	P2	Llega P2
3	P1	P2	
4	P2	P1	
5	P3, P2	P1	Llega P3
6	P1, P3	P2	
7	P1	P2	
8	P1	P3	Fin P2
9	P1	P3	
10	P3	P1	
11	P3	P1	
12	P1, P4	P3	Llega P4
13	P1, P4	P3	
14	P3, P1	P4	
15	P3, P1	P4	
16	P4, P3	P1	
17	P4, P3	P1	
18	P4	P3	Fin P1
19	P4	P3	
20	P3	P4	
21		P3	Fin P4

22		P3	
23		P3	
24			Fin P3

	a)	b)	c)
Tiempo medio de espera	3.5	7.75	7

34. Sea un sistema cuyo algoritmo de planificación de la CPU es Round Robin con cuanto=2. En un instante dado de tiempo llegan dos procesos al sistema A y B cuyas características son:

Proceso	Instante de llegada	Hilos de ejecución	Cómputo
A	0	A0	3
	0	A1	2
	0	A2	1
B	0	B0	4
	0	B1	2

Suponiendo que en dicho sistema los hilos de ejecución son creados y manejados a través de bibliotecas del lenguaje de programación que utiliza un algoritmo de planificación FCFS (por turno de llegada). Realice el cronograma de planificación a nivel de usuario y de núcleo.

Solución

Usuario

A0	A0	B0	B0	A0	A1	B0	B0	A1	A2	B1	B1		
----	----	----	----	----	----	----	----	----	----	----	----	--	--

Núcleo

A	A	B	B	A	A	B	B	A	A	B	B		
---	---	---	---	---	---	---	---	---	---	---	---	--	--

35. Dado un sistema multiprogramado con un único dispositivo de E/S cuyo algoritmo de planificación es FCFS, en el que se tienen cuatro procesos con las siguientes características:

Proceso	Instante de llegada	Intervalos de Cómputo y E/S
A	0	4 CPU, 2 E/S, 4 CPU
B	2	1 CPU, 3 E/S, 3 CPU
C	5	3 CPU, 1E/S, 3 CPU

- a) Indique el orden en que finalizarían los procesos con un algoritmo Round-Robin con un cuanto de tiempo q=4
b) Indique cuál sería el tiempo promedio de espera si se utiliza un algoritmo de planificación de la CPU, SRTF.

Solución

a) A, B, C

b) $(7 + 2 + 2) / 3 = 3.6666$

36. Indica las diferencias entre un proceso y un programa.

Un **programa** es una lista de instrucciones, un **ente pasivo**. Se almacena en ficheros.

Un **proceso** es un programa en ejecución, un **ente activo**. Requiere que el programa se cargue previamente en memoria

37. Indique si las siguientes afirmaciones son verdaderas (V) o falsas (F) :

- Un sistema por lotes multiprogramado y un sistema de tiempo compartido tienen en común que ambos son interactivos. **Falso**

- La multiprogramación permite aprovechar las ráfagas de espera (de la E/S) de un trabajo para ejecutar las ráfagas de CPU de otro trabajo. **Verdadero**

- El sistema operativo se ocupa, solamente, de la gestión de procesos, de la gestión de ficheros y de la gestión de la memoria. De la protección de ficheros y de la gestión de los recursos de E/S se encargan directamente los programas de usuario. **Falso**

- El intérprete de órdenes es la interfaz primaria entre el usuario y el sistema operativo. Se implementa como un programa que lee de su entrada estándar una orden introducida por un usuario, la analiza y la ejecuta. **Verdadero**

- Las instrucciones de E/S se pueden ejecutar en modo supervisor y en modo usuario. **Falso**

38. Dado el siguiente código incompleto, indica qué se debería escribir en los puntos indicados (A, B, C y D), de modo que cada proceso diga quién es (padre o hijo). Además, el proceso padre mostrará el PID de su hijo, mientras que el proceso hijo imprimirá su propio PID.

```
while ( TRUE ) {
    imprimir_prompt();
    leer_orden( orden, params );
    pid = fork();
    if ( pid != 0 ) {
        printf("Soy el A. El pid del B es %d \n",pid);
        waitpid( -1, &status, 0 );
    } else {
        printf("Soy el C, con pid= %d \n", D);
        exec( orden, params, 0 );
        printf( "Mensaje de error\n" );
        exit( 1 );
    }
}
```

Solución:

- a) *padre*
- b) *hijo*
- c) *hijo*
- d) *getpid()*

39. Indique, de entre las siguientes afirmaciones, cuáles de ellas son verdaderas (V) y cuáles falsas (F).

- Un cambio de contexto consiste en desasignar la CPU al proceso en ejecución y asignarla a un proceso preparado, guardando el contexto del proceso en ejecución. **Verdadero**

- El Bloque de Control de Procesos es una estructura de datos en la que se almacenan los atributos de un proceso. **Verdadero**

- Los cambios de contexto implican una sobrecarga importante en el sistema operativo si se hacen con frecuencia. **Verdadero**

- En el PCB se guarda el estado del proceso y su contador de programa, entre otros. No guarda ni el UID ni el GID del usuario/grupo propietario del proceso. Estos datos se guardan en el i-nodo del proceso. **Falso**

40. Indica los tres tipos de planificadores que existen cuando el recurso a asignar es la CPU y en qué se diferencian.

Planificador a largo plazo: selecciona procesos de la cola de procesos que están esperando ser ejecutados y los carga en memoria. Controla el grado de multiprogramación. Se ejecuta con poca frecuencia.

Planificador a medio plazo: se encarga de controlar qué procesos, de entre todos los iniciados deben estar en memoria y qué otros deben estar en el espacio de intercambio. El planificador a medio plazo se encarga de sacar el proceso y volverlo a introducir más tarde.

Planificador a corto plazo: selecciona un proceso de la cola de procesos preparados para ejecución y le asigna la CPU. Se ejecuta con mucha frecuencia, cuando ocurre un evento que conduce a la interrupción del proceso actual, expulsando el proceso a favor de otro.