PLANIFICACION DE LA CPU

PREGUNTA 5.2

En la Planificación Expropiativa se puede observar que un proceso puede ser "expropiado" de la CPU, es decir, es desalojado del mismo para asignárselo a otro proceso de acuerdo a parámetros establecidos en el algoritmo q se está empleando (tiempo de ráfaga, prioridad).

Por otro lado la Planificación No Expropiativa permite que un proceso tenga la CPU a su disposición hasta que este haya concluido, y solo luego de esto se le asigna la CPU al siguiente proceso.

En cuanto a porque sería poco probable que se use una Planificación No Expropiativa en un Centro de Cómputo, podríamos tomar el Centro de Computo 1 de la FIIS como ejemplo de la siguiente manera:

Dicho centro de cómputo tiene su razón de ser como una herramienta primordial para el aprendizaje y desarrollo cognoscitivo de los alumnos, de este modo se ha establecido una serie de horarios en los cuales los alumnos separados en grupos (procesos) pueden acceder al mismo. De este modo se le asigna a cada grupo un determinado tiempo en el centro de computo (CPU), a fin de que luego de que el primer grupo (proceso 1) haya usado todo el tiempo que se le asigno, se le desaloje y el siguiente grupo (proceso 2) pueda hacer uso del centro de computo (CPU), aunque el primero haya terminado o no (expropiación).

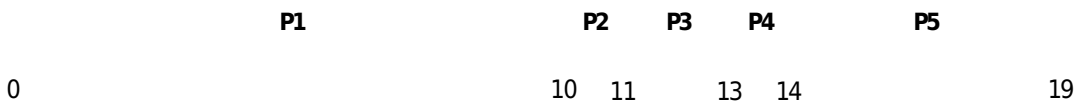
PREGUNTA 5.3

a.

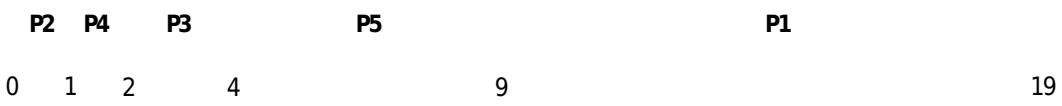
PROCESO	TIEMPO DE RAFAGA	PRIORIDAD
P1	10	3
P2	1	1
P3	2	3
P4	1	4

P5 5 2

a.1 FCFS

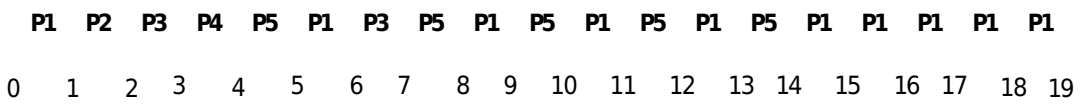


a.2 SJF



a.3 Prioridad no expropiativa

a.4 RR (ROUND – ROBIN)



b. Tiempo de retorno es equivalente al intervalo entre el momento de presentación de un proceso y el momento en que se termina, conocido también como tiempo de servicio.

FCFS

PROCESO	TIEMPO DE PRESENTACION	TIEMPO DE TERMINO	TIEMPO DE RETORNO
P1	0	10	(10 - 0) = 10
P2	0	11	(11 - 0) = 11
P3	0	13	(13 - 0) = 13
P4	0	14	(14 - 0) = 14
P5	0	19	(19 - 0) = 19

SJF

PROCESO	TIEMPO DE PRESENTACION	TIEMPO DE TERMINO	TIEMPO DE RETORNO
P1	0	19	(19 - 0) = 19
P2	0	1	(1 - 0) = 1
P3	0	4	(4 - 0) = 4
P4	0	2	(2 - 0) = 2
P5	0	9	(9 - 0) = 9

Prioridad no Expropiativa

PROCESO	TIEMPO DE PRESENTACION	TIEMPO DE TERMINO	TIEMPO DE RETORNO
P1	0	16	(16 - 0) = 16
P2	0	1	(1 - 0) = 1
P3	0	18	(18 - 0) = 18
P4	0	19	(19 - 0) = 19
P5	0	6	(6 - 0) = 6

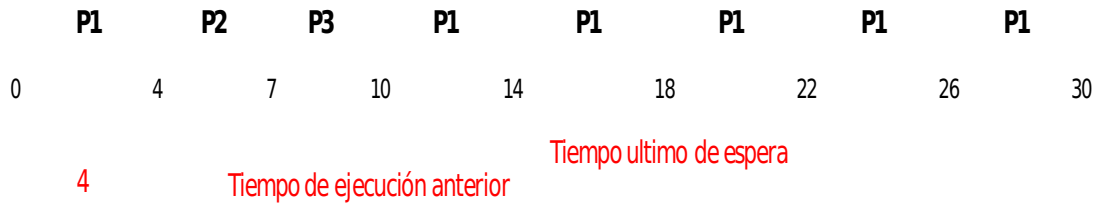
RR (ROUND – ROBIN)

PROCESO	TIEMPO DE PRESENTACION	TIEMPO DE TERMINO	TIEMPO DE RETORNO
P1	0	19	(19 - 0) = 19
P2	0	2	(2 - 0) = 2
P3	0	7	(7 - 0) = 7
P4	0	4	(4 - 0) = 4
P5	0	14	(14 - 0) = 14

c. El Tiempo de espera es la suma de los periodos que el proceso pasa esperando en la cola de procesos listos.

De esta manera lo único que tendríamos que hacer seria, ver el tiempo en que se empezó a ejecutar el proceso. Sin embargo esto no siempre se cumple pues según el ejemplo que propone Silberschatz Galvin de Sistemas Operativos, en el capítulo 5: Planificación de la CPU, de acuerdo a la planificación por Turno Circular (Round – Robin), tenemos:

PROCESO	TIEMPO DE RAFAGA
P1	24
P2	3
P3	3



Donde se afirma que el tiempo de espera del proceso P1 es 6, es decir el tiempo ultimo de espera que vendría a ser 10 el tiempo total en el que ya se estuvo ejecutando anteriormente, que son 4 milisegundos, dándonos así un tiempo de espera de 6 milisegundos.

Entonces podemos concluir que el tiempo de espera se verá afectado cuando hablemos de procesos en los que se uso una planificación expropiativa.

De acuerdo a esto y a los diagramas de Gantt elaborados en la pregunta 1.a, tenemos:

FCFS

PROCESO	**TIEMPO DE EJECUCION ANTERIOR**	**TIEMPO ULTIMO DE ESPERA**	**TIEMPO DE ESPERA**
P1	0	0	(0 - 0) = 0
P2	0	10	(10 - 0) = 10
P3	0	11	(11 - 0) = 11
P4	0	13	(13 - 0) = 13
P5	0	14	(14 - 0) = 14

SJF

PROCESO	TIEMPO DE EJECUCION ANTERIOR	TIEMPO ULTIMO DE ESPERA	TIEMPO DE ESPERA
P1	0	9	(9 - 0) = 9
P2	0	0	(0 - 0) = 0
P3	0	2	(2 - 0) = 2
P4	0	1	(1 - 0) = 1
P5	0	4	(4 - 0) = 4

Prioridad no Expropiativa

PROCESO	TIEMPO DE EJECUCION ANTERIOR	TIEMPO ULTIMO DE ESPERA	TIEMPO DE ESPERA
P1	0	6	(6 - 0) = 9
P2	0	0	(0 - 0) = 0
P3	0	16	(16 - 0) = 16
P4	0	18	(18 - 0) = 18
P5	0	1	(1 - 0) = 1

RR (ROUND – ROBIN)

PROCESO	TIEMPO DE EJECUCION ANTERIOR	TIEMPO ULTIMO DE ESPERA	TIEMPO DE ESPERA
P1	5	14	(14 - 5) = 9
P2	0	1	(1 - 0) = 1
P3	1	6	(6 - 1) = 5
P4	0	3	(3 - 0) = 3

P5	4	13	(13 - 4) = 9

d. ¿Cuál de los planes de la parte a da pie al tiempo de espera promedio más bajo?

PLANIFICACIÓN	**TIEMPO DE ESPERA TOTAL**	**NUMERO DE PROCESOS**	**TIEMPO DE ESPERA PROMEDIO**
FCFS	0+10+11+13+14=**48**	5	9.6 ms
SJF	9+0+2+1+4=**16**	5	**3.2 ms**
Prioridad no expropiativa	9+0+16+18+1=**44**	5	8.8 ms
RR (Round – Robin)	9+1+5+3+9 =**27**	5	5.4 ms

Rpta: El plan SJF da pie al Tiempo de Espera Promedio más bajo.

PREGUNTA 5.4

PROCESO	**TIEMPO DE RÁFAGA**	**PRIORIDAD**
P1	0.0	8
P2	0.4	4
P3	1.0	1

a) Planificación FCFS

PROCESOS	**TIEMPO DE PRESENTACION**	**TIEMPO TERMINO**	**TIEMPO DE RETORNO**

P1	0.0	0.0	0.0
P2	0.4	0.4	0.8
P3	1.0	1.0	1.0
		TOTAL	1.8
		PROMEDIO	**0.6**

b) SJF

PROCESOS	**TIEMPO DE PRESENTACIÓN**	**TIEMPO DE TERMINO**	**PRIORIDAD**	**TIEMPO DE RETORNO**
P1	0.0	0.0	8	0.0
P2	0.4	0.4	4	0.8
P3	1.0	1.0	1	1.0
			TOTAL	1.8
			PROMEDIO	**0.6**

c) Según lo propuesto la CPU estaría ociosa durante la primera unidad de tiempo, es decir que durante un milisegundo de inactividad ningún proceso podría acceder a ella, lo que generaría que los procesos P1 y P2 que se presentan décimas antes de que culmine el tiempo de inactividad de la CPU, aumentarían su tiempo de espera, y su tiempo de retorno, que es el intervalo entre el tiempo de presentación y el tiempo en que se termina el proceso, también sufriría un incremento. A todo ello se suma el hecho de que usaremos la planificación SJF; veamos:

PROCESOS	TIEMPO DE PRESENTACIÓN	TIEMPO OCIOSO DE CPU	TIEMPO DE RÁFAGA	PRIORIDAD	TIEMPO DE TERMINO	TIEMPO DE RETORNO
P1	0.0	1	0.0	8	2.4	2.4
P2	0.4	1	0.4	4	2.4	2.4
P3	1.0	1	1.0	1	2.0	2.0
					TOTAL	6.8
					PROMEDIO	**2.27**

PREGUNTA 5.8

¿Qué relación hay (si acaso existe alguna) entre los siguientes pares de conjuntos de algoritmos?

a) Prioridad y SJF

La relación que existe es que el algoritmo SJF en realidad se puede ver como un algoritmo por Prioridad, si tomamos en cuenta que la prioridad 'p' de sus procesos está asociada o es equivalente a la siguiente ráfaga de CPU, donde a mayor ráfaga de CPU el proceso tendrá una prioridad más baja, y viceversa, cuanto menor sea la ráfaga de CPU más alta será la prioridad del proceso

b) Colas de multinivel con realimentación y FCFS

No existe ninguna relación entre estos dos algoritmos si hablamos de la forma en que trabajan, pues el primero trabaja en una escala mayor que la FCFS. El algoritmo de colas de multinivel con realimentación es capaz de implementar diferentes algoritmos en cada una de las colas e incluso puede hacer que los procesos se intercambien entre las colas según sea conveniente y a fin de usar la CPU de manera eficiente y eficaz. Mientras que el FCFS se limita a procesar según su orden de llegada, sin considerar conveniencia alguna.

c) Prioridad y FCFS

La relación puede observarse en el hecho de que un algoritmo FCFS es equivalente a uno de Prioridad si consideramos que el nivel de prioridad de un proceso para el FCFS está asociado a su orden de llegada. Es decir, un proceso que llega primero tendrá

prioridad absoluta sobre otro que llegue después de él, que tomara posesión de la CPU únicamente cuando el proceso que llego primero termine.

Observación: Hay que la relación será así si y solo si hablamos de una algoritmo por Prioridad del tipo no expropiativa.

d) RR y SJF

No hay ninguna relación entre estos algoritmos pues el primero (RR) trabaja con un *'cuanto'* cantidad de tiempo que le asignará a un proceso para el uso de la CPU, luego del cual desalojará al proceso en ejecución, aún cuando éste no haya concluido (también puede darse el caso de que el proceso en ejecución termine antes del *cuanto*, en cuya circunstancia el proceso liberará voluntariamente el CPU y el planificador se loa asignará al que sigue en la cola); mientras que el segundo no necesita de un *cuanto* y se limita a asignar la CPU al proceso que proceso más corto que encuentre en la cola de procesos listos.