

Sistemas Operativos

Práctico 2

Curso 2024

Ejercicio 1 (Presentación Ejercicio 9) Se tiene un sistema operativo multiprogramado en el cuál se dispone de un único procesador y un modelo de hilos **Mx1**.

Cuenta con un planificador por **prioridades** que sigue las siguientes reglas:

- Cuando un proceso ingresa a la cola de listos, su prioridad será el largo del próximo CPU-Burst (**a nivel de proceso**)
- Cada 5ms, los procesos en la cola de listos suman 5 a su prioridad
- El planificador es **expropiativo**

La planificación a nivel de hilos es **round robin** con un cuanto de 5ms.

Existen dos procesos P1 y P2. P1 a su vez cuenta con dos hilos, P1A y P1B, que ejecutan concurrentemente:

P1		P2
P1A	P1B	
Ejecuta 10ms	Ejecuta 15ms	Ejecuta 10ms
Bloquea 10ms	Bloquea 5ms	Bloquea 10ms
Ejecuta 5ms	Ejecuta 5ms	Ejecuta 10ms

Se asume que la operación de bloqueo lleva un tiempo despreciable, esto podría interpretarse como que la última instrucción del bloque de ejecución se encarga de bloquear.

Inicialmente en el sistema no se tiene ningún proceso de usuario ejecutando y en tiempo $t=0$ se lanza la ejecución de P1 (P1A y P1B) y P2.

Se pide:

- Realice un diagrama de planificación (tiempo vs hilos), comenzando en el tiempo $t=0$, indicando el estado de cada uno de los hilos y su posición en cada una de las colas. Se recomienda utilizar la siguiente notación: Estado_{prioridad} (un proceso en la cola de listos con prioridad 10 L_{10} , un proceso ejecutando E, un hilo en la primera posición de la cola de listos L_1 , etc.).
- Calcule el tiempo de espera y el tiempo de retorno de cada proceso.

Solución: Parte A) Primero debemos analizar el contexto del problema, contamos con un procesador y con un sistema Mx1 (varios hilos de a nivel de usuario a un único hilo a nivel de sistema), esto quiere decir que el sistema solo reconoce un thread de control para el proceso. Entonces tenemos, planificación por threads de usuario. El proceso 1 cuenta con dos hilos, pero para el Sistema Operativo es todo bajo el proceso P1.

El planificador es un planificador por prioridades, por ende también expropiativo y la forma de variación de la prioridad es únicamente por tiempo de espera en la cola de listos (por definición del problema). Entonces:

- $t=0$ ms tenemos que P1 tiene prioridad de 15 (5ms de P1A + 5ms de P1B + 5ms de P1A), y P2 tiene prioridad de 10 (tiempos de CPU burst, dado por la letra)

- P1 empieza a ejecutar y lo hace con P1A, el cuál tiene para ejecutar 10ms, de los cuáles ejecuta 5ms. En ese instante la prioridad de P2 aumenta en 5, llevándolo de 10 a 15. En este momento, la prioridad de P2 = P1 por lo que no expropia al proceso P1, y como tiene hilos (RR de quantum 5ms) el planificador coloca P1B a ejecutar.
- Ejecuta P1B por 5ms y en t=10ms tenemos que la prioridad de P2 es de 20 y la prioridad de P1 es de 15. Por lo que se expropia la CPU, se pone a P2 a ejecutar y P1 entra a la cola de listos. Al entrar a la cola de listos se re-evalúa la prioridad de P1 por su próximo CPU burst que en este caso es de 5ms (esto es lo que le queda a P1A por ejecutar antes de bloquearse). Por lo que la prioridad de P1 es 5.
- Siguiendo con esta idea, P2 ejecuta por 10 ms, ya que la prioridad sube de a 5 cada 5 ms, y luego se bloquea, dejando la CPU a P1, el cuál pone a ejecutar a P1A (ya que es RR y el último de P1 en ejecutar fue P1B).

Siguiendo con este razonamiento tenemos que la planificación queda de la siguiente manera:

		5	10	15	20	25	30	35	40	45	50	55	60	65
Nivel de Usuario	P1A	E	L'1	L'1	L'1	E	B	B	L'1	E	T	T	T	T
	P1B	L'1	E	L'2	L'2	L'1	L'1	L'1	E	L'1	E	B	E	T
Nivel de Kernel	P1	E15	E15	L5	L10	E15	B	B	E15	E15	E15	B	E5	T
	P2	L10	L15	E20	E20	B	B	E10	L5	L10	L15	E20	T	T

En la planificación anterior, L' representa la cola de listos del planificador a nivel de usuario.

Parte B)

Dadas las siguientes definiciones:

Tiempo de Espera: Es la suma de los intervalos de tiempo que un proceso estuvo en la cola de procesos listos (ready queue).

Tiempo de Retorno: Es el intervalo de tiempo desde que un proceso es cargado hasta que este finaliza su ejecución.

Tenemos entonces que pedido en **B)** se reduce a:

- Tiempo de espera P1: $(20-10)=10ms$
- Tiempo de espera P2: $(10-0)+(50-35)=25ms$
- Tiempo de retorno P1: 60ms
- Tiempo de retorno P2: 55ms