

## PRÁCTICO 3

### Objetivos

- Comprender la administración de procesos e hilos de los sistemas operativos
- Comprender la tarea y la elección de un planificador.

### Ejercicio 1 (básico)

Defina que es un proceso a nivel de un sistema operativo.

### Ejercicio 2 (básico)

Varios sistemas operativos populares para microcomputadores ofrecían poco o ningún soporte para el procesamiento concurrente. Comente las principales dificultades que el procesamiento concurrente añade al sistema operativo.

### Ejercicio 3 (básico)

Realice un diagrama de los estados de un proceso y sus transiciones. Explique cuando se produce cada transición.

### Ejercicio 4 (básico)

¿Que es el PCB?

Describe sus campos más importantes

### Ejercicio 5 (básico)

Indicar la secuencia de pasos que sigue el sistema operativo al realizar un cambio de contexto entre dos procesos, teniendo en cuenta la estructura definida en el ejercicio anterior.

### Ejercicio 6 (básico)

Cite dos ventajas que tienen los hilos (threads) sobre los procesos. ¿Qué desventaja importante tienen? Sugiera una aplicación que se beneficiaría del uso de hilos y una que no lo haría.

### Ejercicio 7 (básico)

Describe las acciones del núcleo del sistema operativo (kernel), cuando este conmuta el contexto entre:

- Hilos
- Procesos

### Ejercicio 8 (medio)

¿Qué diferencias hay entre los hilos a nivel de usuario (*user-level threads*) y los hilos a nivel del núcleo (*kernel-level threads*)?

¿En que circunstancias es un tipo mejor que el otro?

**Ejercicio 9 (básico)**

Defina la diferencia entre planificación expropiativa (*preemptive*) y no expropiativa (*non preemptive*). Explique en que ámbitos sería preferible utilizar uno u otro mecanismo.

**Ejercicio 10 (medio)**

¿Qué ventaja tendría definir cuantos de tiempo de diferente tamaño en distintos niveles de un sistema de colas multinivel?

**Ejercicio 11 (avanzado)**

Considere el siguiente algoritmo de planificación por prioridad expropiativo basado en prioridades que cambian dinámicamente. Un número de prioridad mayor implica una prioridad más alta. Mientras un proceso esta esperando la CPU (en la cola de procesos listos, pero no ejecutándose), su prioridad cambia con rapidez  $\alpha$ , cuando esta ejecutándose, su prioridad cambia con rapidez  $\beta$ . Todos los procesos reciben una prioridad de 0 al ingresar en la cola de procesos listos. Los parámetros  $\alpha$  y  $\beta$  pueden ajustarse para dar muchos algoritmos de planificación distintos.

- ¿Qué algoritmo se obtiene si  $\beta > \alpha > 0$ ?
- ¿Qué algoritmo se obtiene si  $\beta < \alpha < 0$ ?

**Ejercicio 12 (medio)**

Suponga que un algoritmo de planificación favorece los procesos que han consumido la menor cantidad de tiempo de procesador en el pasado reciente. ¿Por qué este algoritmo favorecería a los procesos limitados por E/S pero sin postergar infinitamente los procesos limitados por CPU?

**Ejercicio 13 (en clase)**

Se pide implementar un planificador/despachador de procesos para un sistema monoprocesador multitarea que cumpla con las siguientes características:

- Utilice una planificación basada en colas multinivel y sin retroalimentación.
- Los procesos tienen asociada una prioridad discreta, del 0 al 10, siendo la prioridad 0 la mayor del sistema.
- Nunca debe detenerse o retrasarse la ejecución de un proceso con prioridad Q, si solo existen en el sistema procesos con prioridad P, donde  $P < Q$ .
- El quantum es fijo para todas las colas (quantum = q).

**Ejercicio 14 (avanzado)**

Consideremos  $n$  procesos compartiendo la CPU en forma Round Robin. Asumiendo que cada cambio de proceso lleva  $s$  segundos, cual debe ser el time slice (cuanto)  $q$ , para que el tiempo desperdiciado en el cambio de procesos sea minimizado, pero al mismo tiempo se garantiza que cada proceso tendrá control de la CPU por lo menos cada  $t$  segundos.

**Ejercicio 15 (en clase)**

Cinco programas, **A**, **B**, **C**, **D** y **E**, son lanzados a ejecutar en forma simultánea. Los tiempos de ejecución en un ambiente de monoprogramación se estiman en 10, 6, 2, 4 y 8 minutos respectivamente. Las prioridades son 3, 5, 2, 1 y 4 respectivamente, siendo 5 la mayor prioridad. Se desea estimar los tiempos de permanencia en el sistema para cada programa, ignorando el tiempo de intercambio del procesador entre tareas, para los siguientes estrategias de despacho:

- (a) Round robin
- (b) Priority scheduling
- (c) First come, first served
- (d) Shortest job first

Se asumirá para el primer caso que el ambiente es de multiprogramación con una distribución equitativa del CPU. Para el segundo y tercer caso se supondrá que solo ejecutan de a uno por vez, en secuencia. En el tercer caso, asumir en orden de llegada. Por último todos los programas se supondrán acotados por el CPU o sea que no realizan entradas y salidas.

**Ejercicio 16 (medio)**

Sea un sistema monoprocesador multiprogramado y considere el conjunto de procesos siguiente, donde la duración de la ráfaga de CPU se mide en milisegundos:

| Proceso        | Tiempo de ráfaga | Prioridad |
|----------------|------------------|-----------|
| P <sub>1</sub> | 10               | 3         |
| P <sub>2</sub> | 1                | 1         |
| P <sub>3</sub> | 2                | 3         |
| P <sub>4</sub> | 1                | 4         |
| P <sub>5</sub> | 5                | 2         |

Se supone que los procesos llegaron en el orden P<sub>1</sub>, P<sub>2</sub>, P<sub>3</sub>, P<sub>4</sub>, P<sub>5</sub>, todos en el instante 0.

- (a) Dibuje diagramas de Gantt que ilustre quien tiene asignado el procesador y el estado de los procesos en el tiempo, utilizando planificación FCFS, SJF, una técnica por prioridad no expropiativa (a menor numero, mayor prioridad), y RR con cuanto = 1.
- (b) Calcule los tiempos de retorno y espera de los procesos anteriores con cada uno de los algoritmos de planificación de la parte a.
- (c) ¿Cuál de los planes de la parte a da un tiempo de espera promedio mas bajo? (Considerando todos los procesos).

**Ejercicio 17 (medio)**

Sea un sistema que cuenta con los siguientes cuatro procesos con sus respectivos tiempos de ejecución (burst time):

| Proceso        | Tiempo de ejecución |
|----------------|---------------------|
| P <sub>1</sub> | 5                   |
| P <sub>2</sub> | 4                   |
| P <sub>3</sub> | 1                   |
| P <sub>4</sub> | 6                   |

- Realice un diagrama en el tiempo del uso del procesador y el estado de los procesos para los siguientes planificadores: FCFS, SJF y RR con tiempo de quantum 2.
- Calcule el tiempo promedio de espera para los 3 planificadores.
- Realice el diagrama para el planificador RR con tiempo de quantum 5 y haga un análisis de cómo se comporta.

**Ejercicio 18 (en clase)**

Sea un sistema monoprocesador que tiene dos procesos que van a comenzar a ejecutar desde el instante  $t = 0$ .

Estos procesos se comportan de la siguiente manera:

- Ejecutan un bucle durante 50ms.
- Se bloquean durante 100 ms (Por ejemplo, con operaciones de E/S).
- Ejecutan un bucle durante 150ms.

Se pide:

- Dibujar un esquema o diagrama de planificación (tiempo versus procesos), en el que se indique el estado de cada proceso (listo/ejecutando/bloqueado) en cada intervalo de tiempo.
- Calcule el tiempo de espera de los procesos (waiting time).

Nota : El tiempo necesario para realizar un cambio de contexto es 0. Asuma planificación FCFS y Round Robin con quantum de 100ms

**Ejercicio 19 (avanzado)**

Se tiene un sistema operativo simétrico multiprogramado con un planificador con dos colas de prioridad (alta/baja) con retroalimentación (multi-level feedback queue con dos niveles de prioridad). En cada una de las dos colas se utiliza un algoritmo round-robin con quantum de 2 unidades de tiempo.

Los procesos al ser creados se les asignan la prioridad alta, la cual es modificada según los siguientes criterios:

- Un proceso sube a la cola de mayor prioridad si en las últimas 2 unidades de tiempo no ha usado el recurso procesador.
- Un proceso baja a la cola de menor prioridad si en las últimas 2 unidades de tiempo a utilizado completamente el recurso procesador (o sea, agotó su quantum ya que lo uso las 2 unidades).

A su vez, cuando un proceso pasa a un estado bloqueado su quantum es reiniciado.

Sean tres procesos que son creados en el instante  $t=0$  y ejecutan lo siguiente:

1. Ejecutan un bucle durante 1 unidad de tiempo.
2. Se bloquean durante 3 unidades de tiempo.
3. Ejecutan un bucle durante 5 unidades de tiempo.

a. Asumiendo un sistema monoprocesador, realice un esquema o diagrama de planificación (tiempo vs. procesos), en que se indique el estado de cada proceso (listo/ejecutando/bloqueado/terminado) en cada intervalo de tiempo. Además, se debe indicar, en cada caso, el nivel de prioridad (alta/baja) en el que se encuentra el proceso.

b. Asumiendo un sistema con dos procesadores, realice un esquema o diagrama de planificación (tiempo vs. procesos), en que se indique el estado de cada proceso (listo/ejecutando/bloqueado/terminado) en cada intervalo de tiempo. Además, se debe indicar, en cada caso, el nivel de prioridad (alta/baja) en el que se encuentra el proceso.

c. Calcule las siguientes métricas para las partes a y b.

1. Utilización de CPU en los primeras 10 unidades de tiempo y en total (desde el  $t=0$  hasta que el último proceso termina de ejecutar).
2. Tiempo de espera de cada proceso y promedio.

d. Sea un sistema con las siguientes características:

- Tiene disponible cuatro procesadores.
- El sistema operativo soporta solamente el modelo de hilos (threads)  $M \times 1$  (Many-To-One).

Se dispone de un proceso con dos hilos de ejecución y otro proceso con un solo hilo de ejecución.

1. Realice un esquema o diagrama de planificación. Todos los hilos tienen la estructura de ejecución que fue presentada anteriormente (1bucle/3bloqueado/5bucle).
2. Calcule la utilización de CPU en el tiempo total.

Nota:

- Asuma que los tiempos utilizados por el sistema operativo (ej. ejecución del planificador) y los cambios de contexto son nulos.