

Planificación de CPU y Secciones Críticas

Elaborar un programa en el lenguaje de programación C, que intercambie entre los algoritmos de planificación de CPU FSJ apropiativo y no apropiativo (Por cantidad de instrucciones y por Consumo de memoria) y Rond Robin. Para ello se mantendrá un hilo en ejecución que escucha la indicación del Proceso Planificador de Corto Plazo y hará el cambio de Política.

Para ello debe programar las siguientes estructuras:

Una lista de 150 peticiones de procesamiento que se crearán inicialmente y que conformará la Lista de Peticiones Nuevas y que será administrada por el Planificador de Largo Plazo) Cada proceso creado tendrá de esta manera ya su tiempo de creación, para la aplicación de los algoritmos para tiempo de llegada, FSJ no apropiativo será el primero con el que se arranque ejecución. Este tiempo será de carácter aleatorio para cada proceso.

A cada proceso se le debe asignar un valor aleatorio correspondiente a su tamaño de Pila de código es decir cantidad de instrucciones, este será en un rango de 20 000 a 70 000. (Este valor servirá para ejecutar el algoritmo FJS por cantidad de instrucciones)

A cada proceso se le debe asignar un valor aleatorio correspondiente a su consumo de memoria que será un valor aleatorio entre 50 M y 900 M (Este valor servirá para ejecutar el algoritmo FJS por consumo de memoria)

Cada vez que el proceso sea elegido por el planificador de corto plazo para entrar a ejecución este determinará un valor aleatorio entre 10 y 50 que corresponderá a la cantidad de instrucciones que este proceso ejecutará antes de ir a Entrada y Salida.

El planificador de corto plazo tendrá en ejecución un número no menor a 100 procesos en ejecución de manera que cuando baja ese valor deberá ir indicando al planificador de largo plazo cual es el proceso que entrará a ejecución. Por supuesto debe gestionar las recomendaciones al Planificador de Largo Plazo para que ordene su cola según la conveniencia del Planificador de corto.

Cada vez que el proceso sea sacado de ejecución porque se le han terminado sus ciclos, irá a Entrada y Salida, en ese momento se correrá una rutina para asignarle el Tiempo de Espera por Entrada y Salida. (Este tiempo se le asignará cada vez que entre a entrada y salida y será siempre diferente, un valor entre 250 y 1000) Cuando ese tiempo pase, según el reloj cronológico que se definió este pasará a la cola de listos y se ordenará según el algoritmo de turno.

Implementación intercambio automático entre políticas de planificación.

Debe construir:

Bloque de Control de Proceso: Es el registro más importante ya que guarda absolutamente toda la información individual de cada proceso: hora de llegada, cantidad de iteraciones, avance en su pila de código, tiempo de espera en E/S, promedio de tiempo de espera en E/S, tiempo promedio de ejecución, promedio de espera en Listo.

Tabla de Procesos: La tabla de procesos es un registro histórico del comportamiento de los procesos así pues se tienen: rendimiento, tasa de aprovechamiento, tasa de retorno, categorización de procesos por consumo de E/S, las mismas variables arriba citadas, pero como promedios de todos los procesos a ese instante de ejecución. Cada tabla de proceso se debe guardar en un archivo separado.

El Quantum en la ejecución del algoritmo Round Robin es configurable de manera manual pudiendo bajarse o subirse en cualquier momento.

Además, el programa debe ir escribiendo en un archivo de texto toda la ejecución, así como en pantalla, por lo tanto, es necesario en todo momento conocer que algoritmo está en ejecución, cual proceso está en ejecución, cual es la lista de E/S, cual es el orden de Listos.

Pueden existir varios demonios de sistema encargados de cosas específicas:

- Para el intercambio de política del algoritmo a corto plazo
- Cálculo de tiempos promedio
- Configuración de Quantum
- Entrada y Salida
- Estadísticas
- Escritura de los archivos
- Entre otros