

Segunda Práctica

Monitores

La práctica se divide en dos partes diferenciadas:

1. **La resolución teórica:** La primera parte consiste en resolver la práctica como si de un ejercicio de clase se tratase. En la resolución se deben justificar correctamente las decisiones que se tomen. Para la resolución de la misma se deberán utilizar monitores como herramienta para la programación concurrente, además, la semántica para la operación *resume()* será **Desbloquear y Continuar (DC)**. Esto deberá facilitar su posterior implementación en Java. A la finalización de esta parte se deberá entregar un fichero en formato **PDF** con la siguiente estructura: **ApellidoNombrePrac2.pdf**. En ILIAS encontraréis la entrega en el sistema antiplagio de la Universidad. Un porcentaje mayor del 10% tendrá como consecuencia la no evaluación de la práctica.
2. **La implementación:** Para la implementación tendréis que diseñar una clase que se comporte como un monitor, ya que en Java no dispone de esa herramienta de programación. Para ello se tendrá que elegir las herramientas disponibles en la biblioteca de concurrencia. El marco de ejecución de hilos será **Executor** con alguna de las clases que implementan dicha interface. Una vez finalizada la práctica deberá subirse el proyecto NetBean completo en formato **ZIP** con la siguiente estructura: **ApellidoNombrePrac2.zip**. El código se someterá a la detección de copias y un porcentaje mayor del 10% tendrá como consecuencia la no evaluación de la práctica.

Práctica a Realizar

Tenemos un sistema que cuenta con una memoria principal que dispone de un total de **NUM_MARCOS (20)** marcos para las páginas de los procesos. Hay que diseñar los siguientes elementos del sistema:

- **Un monitor para el control de acceso a memoria:** Será el encargado de centralizar las operaciones que deben realizar los procesos para operar con la memoria del sistema para los procesos y el gestor de memoria.
 - Ningún proceso puede tener asignados más de 4 marcos de memoria.
 - Para la petición inicial de un proceso se le deben asignar siempre 2 marcos disponibles. Si solo queda un marco disponible se puede asignar a un proceso que tenga un fallo de página.
 - Si no se le pueden asignar más marcos se sustituirá la página que lleve más tiempo cargada en memoria para ese proceso.
- **Los procesos:** Para simular el comportamiento de los procesos en el sistema diseñaremos una tarea que tenga el siguiente comportamiento:
 - Solicitará al monitor que se le asignen los marcos al inicio de su ejecución. Inicialmente se le asignan dos marcos para las páginas 1 y 2 del proceso.
 - Realizará un número variable de ciclos de ejecución, entre 8 y 12. Ese número simula las instrucciones del proceso para completar su ejecución.
 - Para cada ejecución, que se simulará por un tiempo de entre 1 y 2 segundos:
 - Genera un número de página de entre los disponibles del proceso.
 - Pregunta al monitor si la página está cargada en memoria.
 - Si la página no está cargada en memoria se solicitará al monitor que cargue la página en un marco de memoria.
 - Cuando se complete el último ciclo finalizará el proceso. Antes de finalizar deberá comunicar al monitor que libera los marcos de memoria que tenga asignados.
- **Gestor de Memoria:** Se deberá crear una tarea para simular el comportamiento del gestor de memoria.
 - Comprueba en el monitor las peticiones pendientes de liberación de marcos. La petición no puede bloquear al gestor de memoria si no hay peticiones pendientes.
 - Atiende todas las peticiones pendientes.
 - Comprueba en el monitor las peticiones pendientes de marcos que han realizado los procesos al monitor. La petición no puede bloquear al gestor de memoria si no hay peticiones pendientes.
 - Atiende una petición cada vez.

- **El sistema:** La simulación del sistema se realizará mediante el programa principal y tendrá el siguiente comportamiento:
 - Creará y ejecutará el gestor de memoria como primer proceso.
 - Creará y ejecutará un nuevo proceso en un intervalo comprendido entre 1 y 3 segundos. Para cada proceso se generará un número aleatorio entre 4 y 8 que indicará el número de páginas de memoria que representa el tamaño total del proceso.
 - El tiempo de ejecución del sistema será de 3 minutos. Pasado ese tiempo finalizará los procesos que aún están activos y el gestor de memoria.
 - El sistema presentará los siguientes datos de cada proceso que haya concluido su ejecución:
 - Tiempo de inicio del proceso.
 - Tiempo de ejecución del proceso.
 - Número de fallos de página del proceso.
 - Mostrará el número de procesos que no han concluido su ejecución.
 - La asignación media de marcos a los procesos.
 - La media de fallos de página de los procesos.

NOTA: El inicio del proceso es cuando solicita al monitor la asignación inicial de marcos. La finalización del proceso es cuando completa la solicitud al monitor para devolver los marcos asignados. Se deben completar las peticiones de memoria que realicen los procesos, una vez iniciada su ejecución, aunque no haya marcos disponibles.