

Programación Concurrente y de Tiempo Real

Grado en Ingeniería Informática

Asignación de Prácticas Número 7

En esta asignación aplicará control de exclusión mutua y sincronización, utilizando para ello el API estándar de Java, efectuando la implementación con monitores a soluciones de problemas clásicos de la concurrencia. **Documente todo su código con etiquetas (será sometido a análisis con javadoc).**

1. Enunciado

En el tema tres del curso teórico se ha analizado el concepto de monitor, y se ha resuelto el problema del productor-consumidor con él; a partir de aquí:

- repase exhaustivamente el funcionamiento del concepto de monitor, y del monitor productor-consumidor desde un punto de vista teórico (ver tema 3 del curso de teoría).
- a continuación:
 - escriba en `prodCon.java` una versión en Java del monitor para el problema del productor-consumidor donde: modelará el recurso compartido (*buffer*) mediante un *array* de enteros, tendrá todos sus métodos en exclusión mutua y utilizará sincronización `wait()`-`notifyAll()`, empleando **siempre** condiciones de guarda.
 - escriba ahora un diseño de hebras productoras y consumidoras que utilicen este monitor mediante herencia de la clase `Thread`, y guárdelo en `usaprodCon.java`.
 - reduzca el tamaño del buffer a uno y vea qué ocurre si únicamente hay un productor y un consumidor.
 - ejecute un productor y varios consumidores y vea qué ocurre.
 - ejecute varios productores y un consumidor y vea qué ocurre.
- redacte un documento `analisis.pdf` con sus impresiones acerca de las diferentes pruebas de ejecución propuestas.

2. Procedimiento de Entrega

PRODUCTOS A ENTREGAR

- Ejercicio 1: `prodCon.java`, `usaprodCon.java` y `analisis.pdf`.

MÉTODO DE ENTREGA: Tarea de Moodle.