

780014 - PROGRAMACIÓN AVANZADA

Examen extraordinario de evaluación – 3 de julio de 2014

Normas de Examen:

La duración del examen es de 3h.

Se numerarán todas las hojas con el número de hoja y el total (ejemplo: 1/4, 2/4, 3/4, 4/4).

El alumno se podrá quedar con el enunciado del examen.

Parte 1: Paradigmas, programación concurrente, y concurrencia de memoria común

1. - (1 punto) Explicar brevemente la relación entre la evolución de los lenguajes y la aparición de los paradigmas de programación.

2. - (1 punto) Explicar brevemente las distintas versiones de Colas concurrentes basadas en la interfaz BlockingQueue.

3. - (3 puntos) Una **Agencia** de servicios turísticos funciona de la siguiente manera:

- Dispone de 10 minibuses con capacidad para 8 viajeros.
- Cuando un viajero quiere visitar Alcalá, se pone a la cola para montar en un minibus.
- A la puerta de la agencia se pone un minibus libre que espera hasta que se completa con 8 viajeros. Entonces realiza su recorrido de 10-15 segundos y a continuación se bajan los viajeros, quedando libre para otro viaje.
- En cuanto parte uno, se pone otro libre a la puerta a esperar nuevos viajeros. Si los diez minibuses están ocupados, los viajeros esperan en la cola.
- Cuando un viajero descende de un minibus, da un paseo de 30 a 80 segundos y vuelve a la puerta de la agencia para otro viaje.

Los viajeros son threads de la clase **Viajero** que llegan en momentos al azar a la **Agencia** y ejecutan el método *verAlcala()*.

Se pide escribir la clase **Agencia** con dicho método y todo lo necesario para seguir el funcionamiento descrito.

También se pide escribir la clase **Viajero** y un programa **Principal** que cree 50 viajeros y los active de uno en uno cada 5-10 segundos.

Por último, explicar cómo funciona el sistema y las herramientas de sincronización usadas.

Parte 2: Programación concurrente distribuida

4. - (1 punto) Explicar las clases implicadas en la creación de una aplicación con Sockets en Java de tipo UDP y de tipo TCP.

5. - (1 punto) Explicar y mostrar en cronogramas la diferencia entre el comportamiento síncrono y asíncrono de las aplicaciones distribuidas.

6. - (3 puntos) **Crear** un programa utilizando RMI que permita a una serie de estaciones meteorológicas enviar lecturas de temperatura, velocidad del viento y humedad.

El programa receptor, deberá poder atender a todos los emisores necesarios y se limitará a imprimir en pantalla las lecturas recibidas, reuniendo todas las lecturas recibidas con la misma hora, minuto y segundo antes de imprimirlas y sin implementar ningún comportamiento que pueda mostrar espera activa.

Los programas emisores enviarán continuamente los 3 datos requeridos y la fecha y hora en que han sido medidos. Los datos serán simulados mediante la generación aleatoria y la marca de fecha se tomará en el sistema de cada programa emisor. El envío de lecturas se hará cada 3-7 segundos (con un factor aleatorio).