
PARADIGMAS DE LA PROGRAMACIÓN

Práctica de Laboratorio (E1)

Convocatoria Ordinaria – 2022/2023

Publicado el 08 de noviembre de 2022

Simulación del funcionamiento de un sistema de reparto de cartas

Programación Concurrente

Se desea modelar el comportamiento de un grupo de 400 personas (que serán modeladas como hilos) que van dejando, cada una de ellas, dos cartas en un buzón. Cada persona tardará entre 400 y 800 ms entre dejar una carta y otra. La capacidad máxima del buzón es de 30 cartas. Por otro lado, hay 5 empleados (que también deberán ser modelados como hilos) de la empresa de cartas que se encargan de coger cada una de las cartas del buzón y depositarlas en dos furgonetas de reparto. Debido a las características de cada furgoneta, sólo un empleado puede entrar a depositar una carta a la vez en una determinada furgoneta. La primera carta de cada persona va a la primera furgoneta, y la segunda carta de cada persona va a la segunda furgoneta.

Las personas sólo podrán dejar las cartas en el buzón mientras haya hueco disponible, y si no deberán esperar a que lo haya. Por otro lado, los empleados deberán esperar a que haya cartas disponibles en el buzón para poderlas llevar a las distintas furgonetas. Los empleados tardarán entre 400 y 700 ms desde que cogen una carta hasta que la llevan a una furgoneta, y el mismo tiempo en volver al buzón.

Consideraciones a tener en cuenta:

- Cada persona deberá tener un identificador único: “Persona1”, “Persona2”, etc.
- Los empleados tendrán identificadores “Empleado1”, “Empleado2”, etc.
- Una vez que una persona ha depositado las dos cartas en el buzón, finaliza su ejecución.
- Los empleados no finalizarán nunca su ejecución.
- Suponemos que la capacidad de las furgonetas es ilimitada.
- Cada carta estará identificada de la siguiente manera: “IDdeSuPersona+Cx”. Por ejemplo, las cartas de la “Persona5” tendrán como identificador: “Persona5-C1” y “Persona5-C2”.

Los tiempos de los pasajeros y empleados se generarán aleatoriamente mediante las funciones random de Java, y todo el comportamiento del sistema se guardará en un log (un fichero de texto llamado

“evolucionCartas.txt”), además de mostrarse gráficamente por pantalla, de forma que sea sencillo analizar lo sucedido. El log guardará los eventos que van teniendo lugar, por ejemplo: “Persona1 deja la carta Persona1-C1 en el buzón”, “Empleado2 coge la carta Persona1-C1 y la lleva a la furgoneta 1”, etc. En cada línea de dicho log deberá constar la marca de tiempo (incluyendo el segundo determinado en el que tuvo lugar el evento) y el evento en sí.

Una posible interfaz del sistema sería la que se puede apreciar en la siguiente imagen:

The screenshot shows a window titled "Sistema de Cartas". It contains the following elements:

- Mailbox Section:**
 - Label: "Contenido del buzón:"
 - Label: "Número de cartas en el buzón:" with a text box containing the value "4".
 - A large text box displaying: "Persona2-C1, Persona3-C2, Persona5-C1, Persona4-C1".
- Employees Section:**
 - Five rows, each with a label "Empleado X:" and a text box:
 - Empleado 1: "Llevando Persona1-C2 a la furgoneta 2"
 - Empleado 2: "Volviendo al buzón de cartas"
 - Empleado 3: "Llevando Persona1-C1 a la furgoneta 1"
 - Empleado 4: "Pausado"
 - Empleado 5: "Volviendo al buzón de cartas"
- Trucks Section:**
 - Label: "Contenido de la furgoneta 1:" with a text box containing "Persona3-C1".
 - Label: "Contenido de la furgoneta 2:" with a text box containing "Persona2-C2".
- Buttons:**
 - A row of three buttons: "Pausar/Reanudar todo", "Pausar/Reanudar Empleado 1", "Pausar/Reanudar Empleado 2".
 - A second row of two buttons: "Pausar/Reanudar Empleado 4", "Pausar/Reanudar Empleado 5".

Se deberá incluir un botón para pausar/reanudar toda la ejecución el programa, así como cinco botones independientes para pausar/reanudar a cada uno de los empleados.

Programación Distribuida

Basándose en la aplicación indicada en el apartado anterior, incluir un nuevo módulo capaz de visualizar de forma remota el contenido del buzón y de las furgonetas (utilizando programación concurrente distribuida). Este nuevo módulo permitirá visualizar, de forma remota y mediante interfaz gráfica, el estado del buzón y de las furgonetas.

Un posible ejemplo de interfaz gráfica es el que se muestra en la siguiente figura.

Módulo de Visualización

Contenido del buzón: Número de cartas en el buzón: 4

Persona2-C1, Persona3-C2, Persona5-C1, Persona4-C2

Contenido de la furgoneta 1:

Persona3-C1

Contenido de la furgoneta 2

Persona2-C2

Se deben desarrollar, en total, dos programas:

- Un servidor, cuyo código base será el programa desarrollado en el apartado “Programación Concurrente”, ampliado con la funcionalidad correspondiente para dar soporte al módulo de visualización de programación distribuida.
- Un programa cliente (módulo de visualización) que permita comprobar el contenido del buzón y de las dos furgonetas de forma remota.

Se podrán utilizar todos los mecanismos vistos en clase para resolver todos los problemas de comunicación y sincronización que se plantean en este enunciado. No obstante, se deben utilizar los mecanismos de sincronización y comunicación que resuelvan el problema **de la forma más eficiente y óptima** posible.

Condiciones de entrega

1. La práctica se realizará (*opcionalmente*) *por parejas* y deberá ser entregada **antes** de la fecha indicada en el Aula Virtual, a través de la tarea correspondiente, mediante la subida de dos archivos: la memoria de la práctica en formato PDF o DOC y el proyecto Netbeans completo,

comprimido como **ZIP** (no utilizar extensión .rar). No se aceptarán trabajos enviados pasada la fecha límite de entrega.

2. La entrega fuera del plazo indicado en el Aula Virtual supondrá una reducción en la calificación final, siendo del 25% si se entrega el día siguiente a la fecha límite, o del 50% si se entrega dentro de los dos días siguientes. La entrega más allá de esos dos días no será admitida bajo ninguna circunstancia.
3. El proyecto entregado deberá ser un **proyecto de NetBeans**. No se admitirán proyectos realizados con otros entornos de desarrollo.
4. Para aprobar, es condición necesaria que todos los programas funcionen correctamente y de acuerdo a las especificaciones indicadas en los enunciados.
5. Se debe desarrollar la solución haciendo uso de buenas prácticas de programación. Por ejemplo, es necesario que todos los nombres de las clases comiencen por una letra mayúscula y todos los nombres de atributos y métodos comiencen por una letra minúscula; los atributos deberán ser privados, y sólo se podrá acceder a ellos mediante métodos getter y setter.
6. **Ambas partes** (concurrente y distribuida) de la práctica de laboratorio **se deberán entregar juntas** (es decir, en un único proyecto y una única memoria), ya que la parte de programación distribuida se construye sobre la parte concurrente.
7. Si la práctica es realizada por una pareja, **sólo uno de los integrantes deberá subirla** al aula virtual, indicando el nombre de ambos alumnos.
8. En la portada de la memoria deberán figurar los datos siguientes:
 - a. **Grado en Ingeniería en Sistemas de Información**
 - b. **Curso 2022/2023 – Convocatoria Ordinaria**
 - c. **DNI – Apellidos, Nombre**
9. **La memoria deberá incluir, como anexo, el código fuente del programa. Si esto no fuera así, la práctica no podrá ser aprobada.**
10. La memoria explicativa de la práctica realizada deberá incluir, en el orden siguiente: 1) un análisis de alto nivel; 2) diseño general del sistema y de las herramientas de sincronización utilizados; 3) las clases principales que intervienen con su descripción (atributos y métodos); 4) un diagrama de clases que muestren cómo están relacionadas; y 5) el código fuente, como anexo.
11. Dicha documentación, exceptuando el código, no deberá extenderse más de 20 páginas. La calidad de la documentación – presentación, estructura, contenido, redacción – será un elemento básico en la evaluación de la práctica.
12. De cara a la defensa/examen de la práctica, el estudiante podrá llevar impresa una copia de la memoria entregada con la práctica.
13. La resolución de la práctica debe ser genuina y realizada desde cero, es decir, no se podrá utilizar ningún tipo de código fuente de prácticas presentadas a esta convocatoria o en convocatorias anteriores, bien sean propias o de otros compañeros. En caso de detectarse esta situación, la práctica no será evaluada y tendrá una calificación de 0 Suspenso.