

**Laboratorio de Programación Avanzada**  
**PROGRAMACIÓN CONCURRENTE Y DISTRIBUIDA**  
**Práctica de evaluación Junio 2005**

---

## **ENUNCIADO.**

---

### **Parque de atracciones**

El responsable de un parque de atracciones, ha encargado un programa que permita simular el funcionamiento del parque y comprobar el efecto que algunas decisiones producen sobre las colas que se forman en las atracciones, como duración del viaje, puesta en revisión o cierre por mantenimiento, etc.

Para ello hay que construir un modelo en el que el parque está representado por un conjunto de atracciones (recursos compartidos) y los clientes del parque son procesos concurrentes (threads) numerados, que llegan a la puerta del parque, entran cuando el aforo lo permita, utilizan un número variable de atracciones y finalmente abandonan el parque.

El sistema deberá estar dividido en dos partes distribuidas comunicadas entre sí por RMI.

El módulo **Parque** (“servidor”) deberá tener una interfaz gráfica que permita ver en una única ventana el estado de todo el parque y la situación de cada uno de los clientes del parque. Incluirá también dos botones que permitan **detener** y **reanudar** la actividad, con objeto de poder hacer comprobaciones sobre el funcionamiento del sistema.

El módulo **Control** (“cliente”) también deberá tener una interfaz gráfica que permita visualizar el número de usuarios que están en la cola de cada atracción en un momento determinado. Un botón de “**refrescar**” permitirá actualizar dicha información. También incluirá otros dos botones: uno para “poner en mantenimiento” o **cerrar** la noria y otro para volverla a **abrir**.

#### **Funcionamiento:**

1. El número total de clientes que llegan al parque será de 200. Cada uno de ellos estará identificado por un número entre el 101 y el 300. A la puerta del parque irá llegando uno cada 0,1 segundos.
2. El aforo del parque es de 100 clientes, por lo que alcanzado este límite, los que llegan deberán esperar en la cola de la puerta hasta que alguno de los que hay dentro, salga del parque.
3. Dentro del parque hay cuatro atracciones: **noria**, **tiovivo**, **espejos** y pasaje del **terror**. Además, los clientes pueden pasear y descansar en el **recinto** del parque.
4. El número máximo de clientes (capacidad) que puede haber en las atracciones, es el siguiente: noria: 8; tiiovivo: 10; espejos: 15; terror: 20; recinto: sin límite.
5. El funcionamiento de las atracciones es de dos tipos:

- En la noria y el tiovivo, entran tantos clientes como quepan, están un tiempo fijo funcionando y salen de la atracción para que entren otros que estén en la cola. En el caso de que no se alcance la capacidad máxima de la atracción, pero tampoco quede nadie más en la cola, la atracción se pondrá en marcha sin esperar más clientes. (Lógicamente, está prohibido subirse en marcha).
  - En los espejos y el pasaje del terror, los clientes entran, están un tiempo aleatorio (entre 0,2 y 0,8 segundos) y salen. Si se alcanza la capacidad máxima permitida, los clientes que llegan tienen que esperar en la cola y van entrando a medida que salen los que estaban dentro.
6. La noria puede cerrarse, para su revisión, desde el módulo de control. Cuando esto ocurra, el sistema pondrá el cartel de **cerrado**, esperará a que finalice el turno que esté funcionando en ese momento y echará a todos los clientes que se encuentren esperando en la cola. Con el cartel de cerrado, ningún cliente se pondrán en la cola. Cuando, desde el módulo de control, se vuelva a abrir la noria, se quitará el cartel de cerrado y volverá a funcionar normalmente.
  7. El comportamiento de cada cliente será el siguiente: llega a la puerta del parque y si puede entrar lo hace; si no, espera en la **cola de la puerta**. Una vez dentro, se dirigirá a cualquier atracción (que no esté cerrada), esperará turno y la usará. Al salir, se quedará un tiempo aleatorio (entre 0,5 y 1,5 segundos) deambulando por el recinto y después volverá a otra atracción o saldrá del parque.
  8. Todos los clientes entrarán una sola vez en los espejos y en el terror. En la noria (si no está cerrada) y en el tiovivo, pueden no montar o hacerlo hasta un máximo de 5 veces. El orden del uso de las atracciones será completamente al azar. (A la salida del parque, cada cliente escribirá en la pantalla de texto su identidad y el número de veces que ha usado cada atracción)
  9. En todo momento deberá ser visible en la pantalla del módulo Parque la lista de clientes que hay en cada sitio: atracción, cola de cada atracción, cola de la puerta y recinto.
  10. La duración de los viajes en la noria y el tiovivo, no está fijada de antemano. Una vez construido el modelo, se harán pruebas variando la duración de cada una, hasta conseguir unos valores que cumplan con el objetivo de que las colas que se forman no sean excesivas.

### Módulo de Control:

11. Tendrá a su disposición tres métodos remotos ofrecidos, mediante RMI de Java, por el módulo servidor Parque:
  - Un método para la consulta del estado de las colas de las atracciones, **estadoColas** que devolverá el número de clientes que están haciendo cola en cada atracción.
  - Un método **cerrarNoria** para hacer que esa atracción sea puesta en revisión, como se detalla en el apartado 6.
  - Un método **abrirNoria** para que la noria vuelva a su estado normal de funcionamiento.
12. El módulo de control deberá poder arrancarse en un ordenador diferente del que ejecuta el módulo Parque. Ofrecerá una interfaz gráfica con tres botones: “Abrir Noria”, “Cerrar Noria” y “Refrescar” (correspondientes a cada uno de los métodos remotos).
13. Cada vez que se pulse el botón “Refrescar”, se actualizará la información sobre el tamaño de las colas de espera en las atracciones, sin que dicha información se esté mostrando de forma continua.

## Condiciones de entrega

1. La práctica se realizará en grupos de 2 personas como máximo.
2. Es condición necesaria para aprobar la práctica, que todos los programas funcionen correctamente y de acuerdo a las especificaciones indicadas en los enunciados.
3. Deberá entregarse un disquete o CD-ROM debidamente etiquetado y conteniendo una carpeta - cuyo nombre será el DNI de uno de los integrantes del grupo - en la que deberán incluirse un fichero:
  - **Parque.doc o Parque.pdf**, con la memoria de la práctica, como se describe en el apartado siguientey dos carpetas:
  - **Fuentes**, con todos los archivos .java
  - **Clases**, con los archivos .class.Para mostrar el funcionamiento de los programas, será necesario instalarlos en alguno de los ordenadores del laboratorio.
4. Un documento impreso con la memoria explicativa de la práctica (análisis de alto nivel y diseño del sistema) y que contendrá, entre otros, la relación entre clases (jerarquías) con su descripción (atributos y métodos) así como los posibles diagramas que aclaren el análisis y diseño de la aplicación. También deberá contener una descripción de la interfaz implementada y un manual de usuario. El código fuente no se incluirá en la memoria de la práctica
5. En la portada de la memoria, se incluirá:

**Ingeniería Técnica en Informática de \_\_\_\_\_”**  
**Laboratorio de Programación Avanzada. Curso 2004/05**  
**Práctica de P. Concurrente y Distribuida: PARQUE DE ATRACCIONES**  
**DNI – Apellidos, Nombre**  
**DNI – Apellidos, Nombre (si se hace entre 2)**

6. La práctica deberá entregarse, al profesor asignado, el día oficial del examen de la asignatura (21 de Junio de 2005). Cada profesor, establecerá un sistema de reservas de día y hora para la defensa de la práctica.
7. En caso de aprobado, **la calificación obtenida en esta práctica se mantendrá exclusivamente hasta la convocatoria de Septiembre de 2005.**