

Programación Concurrente y de Tiempo Real

Grado en Ingeniería Informática

Asignación de Prácticas Número 11

Se le plantean a continuación algunos ejercicios de programación distribuida con el *framework* Java-RMI. Debe desarrollar los programas que se piden. Documente todo su código con etiquetas (será sometido a análisis con `javadoc`).

1. Enunciados

1. Se desea disponer de un sistema remoto para jugar a la BonoLoto. El servidor genera aleatoriamente seis números entre 1 y 49 que deberán ser acertados. Los clientes envían una apuesta al servidor con un array que incluye los seis números que definen la apuesta. El servidor contesta al cliente si acertó o no con su apuesta. Implemente una arquitectura RMI que permita jugar a la BonoLoto en red para estas especificaciones. Guarde su interfaz en `iBonoLoto.java` y su servidor y cliente en `cBonoLoto.java` y `sBonoLoto.java`.

2. Se desea efectuar el cálculo remoto de la aproximación a π mediante el ya conocido método de Monte-Carlo. Para desarrollar esta aproximación distribuida, un cliente RMI enviará una petición remota al servidor que efectúa el cálculo, indicándole con cuántos puntos contribuye al mismo. Diferentes cliente podrán, mediante sus respectivas llamadas al servidor, contribuir a mejorar la aproximación remota, que será única y de precisión creciente. El servidor deberá ofrecer también un método de `reset()`, que cualquier cliente podrá invocar para reiniciar el cálculo. Escriba una arquitectura RMI completa con ficheros `iPiMonteCarlo.java`, `cPiMonteCarlo.java` y `PiMonteCarlo.java`.

2. Procedimiento de Entrega

PRODUCTOS A ENTREGAR

- Ejercicio 1: `iBonoLoto.java`, `cBonoLoto.java` y `sBonoLoto.java`
- Ejercicio 2: `iPiMonteCarlo.java`, `cPiMonteCarlo.java` y `PiMonteCarlo.java`

MÉTODO DE ENTREGA: Tarea de Moodle.