

Ejercicios Tema 13. Hilos de ejecución

Todos los ejercicios de este boletín deben estar dentro de un **package** llamado **tema13** (Siguiendo las reglas que hemos visto en clase, ej: `com.germangascon.tema13`).

1. Crea un programa que lance un hilo de ejecución (sin esperar a que finalice) que muestre los primeros 100 números enteros. El programa al finalizar debe mostrar "fin de programa".
2. Crea un programa que lance un hilo de ejecución (esperando a que finalice) que muestre los primeros 100 números enteros. El programa al finalizar debe mostrar "fin de programa".
3. Crea un programa que lance un hilo de ejecución en modo demonio (sin esperar a que finalice) que muestre los primeros 100 números enteros. El programa al finalizar debe mostrar "fin de programa". ¿Qué ocurre?
4. Crea un programa que lance un hilo de ejecución en modo demonio (esperando a que finalice) que muestre los primeros 100 números enteros. El programa al finalizar debe mostrar "fin de programa". ¿Qué ocurre?
5. Crea un programa que cree n hilos de ejecución, donde n será un número recibido como parámetro al iniciar el programa (args de la función main). Cada hilo de ejecución tendrá un identificador único asignado en el momento de la creación. Al iniciarse cada hilo mostrará los primeros 10000 números de la siguiente forma.

"Hilo: " + IDENTIFICADOR + " (" + numero + ")\n"

¿En qué orden se ejecutan los hilos?
6. Crea el mismo programa del ejercicio anterior pero esta vez n será el número de núcleos de la CPU de tu ordenador.
7. Crea una función que permita determinar si un número está en un array. No nos interesa su posición, solo saber si se encuentra en el array. Para resolver el problema utiliza hilos de ejecución de forma que cada hilo observe una parte del array. El número de hilos debe ser igual al número de núcleos de tu CPU si el tamaño del array lo permite.
8. Crea una función que cuente la cantidad de números primos que hay en un array. Para resolver el

problema utiliza hilos de ejecución de forma que cada hilo cuente una parte del array. El número de hilos debe ser igual al número de núcleos de tu CPU si el tamaño del array lo permite.

9. Crea un programa que siguiendo el modelo Produtor-Consumidor vaya generando números aleatorios. El número de productores y consumidores debe ser configurable por el usuario, así como el máximo y mínimo número de números generados que debe de haber en todo momento. Los productores generarán números enteros aleatorios, hasta alcanzar el máximo, y almacenándolos en un ArrayList. Los consumidores irán extrayendo números del ArrayList, si el número de elementos del ArrayList baja del mínimo, deberán solicitar a los productores que fabriquen más números aleatorios.