

EJERCICIOS PROGRAMACIÓN MULTITHREAD

1. Crea un hilo que en su método run() muestre un “NO” hasta un máximo de 30 veces. En el método principal (main) tras ejecutar el start() de tu hilo, mostrará “YES” hasta un máximo de 30 veces. Como salida obtendrás una serie alternativa de NOes YESes ya que una vez iniciada la ejecución del thread, el tiempo de la CPU se reparte entre todos los procesos y threads del sistema, con lo cual se intercalan instrucciones del método main() con instrucciones del método run()).

Posible salida:

YES YES YES YES YES YES YES YES YES YES YES YES NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO NO
NO NO NO NO NO YES YES YES YES YES YES YES YES YES YES YES YES YES YES YES YES YES

2. Vamos a hacer un programa similar al anterior pero instanciando 2 threads.
En el HiloA mostrará el mensaje NO, hasta un máximo de 30 veces
En el HiloB mostrará el mensaje YES, hasta un máximo de 30 veces
En el programa principal mostrará un mensaje “Ejecución en HiloA” y se ejecutará el HiloA; luego mostrará un mensaje “Ejecución en HiloB” y se ejecutará el HiloB; y luego un mensaje “Ejecución en main”. Por ejemplo.
3. Hacer un programa que calcule el factorial y un número de la sucesión de Fibonacci. Cada uno de los cálculos tiene que ser realizado por un hilo independiente. Las clases que implementen cada hilo tienen que llamarse Factorial y Fibonacci. Implementar los procesos de 2 maneras diferentes:
 - a) Heredando de la clase Thread
 - b) Implementando la interface Runnable

Recuerda que $\text{factorial}(0)=1$ y para todo $n>0$, $\text{factorial}(n)=n*\text{factorial}(n-1)$.

Recuerda que $\text{fib}(0)=0$, $\text{fib}(1)=1$ y para todo $n>1$, $\text{fib}(n)=\text{fib}(n-1)+\text{fib}(n-2)$.

4. Hacer un programa que calcule los factoriales del 5 al 14 modificando la clase Factorial del ejercicio anterior, para que se visualicen los mensajes:
- empieza el proceso ... cálculo del factorial de: XX*
acabó el proceso el factorial de: XX es XXXX
- En el método main declarar un vector o array de 10 hilos y lanzarlos a ejecución.