

1. Vamos a ver como una variable “contador” no funciona correctamente debido a que se accede concurrentemente a una misma zona de memoria sin que se produzca exclusión mutua (esto intentaremos solucionarlo más adelante cuando se trate el sincronismo de los hilos).

Crea una subclase Thread la cual tendrá un atributo de clase¹ que es una variable entera (contador) que se inicializa a 0 y un atributo de objeto² que es una cadena; como métodos: se redefine su constructor pasándole un string como argumento, se llama al constructor de la superclase, se le asigna al atributo de objeto como valor el del string pasado como argumento.

El método run() mostrará hasta un máximo de 10 veces el valor de contador incrementado y luego el valor del atributo de objeto (el string).

En el programa principal (main) → creas 2 objetos de tu subclase Thread (2 hilos) cada uno con un valor, por ejemplo, el hilo1 con SI y el hilo2 con NO.

Los hilos deberán pertenecer a un grupo denominado "Grupo PSP".

Ejecutas ambos hilos y comprueba que el contador no funcionó todo lo bien que debía ya que no se ha incrementado secuencialmente (como sí tenía que hacerlo).

Por ejemplo:

Salida 1

Comienzo de la ejecución concurrente

1:SI 2:SI 3:SI 4:SI 5:SI 6:SI 7:SI 8:SI 9:SI 10:SI 11:NO 12:NO 13:NO 14:NO 15:NO 16:NO 17:NO 18:NO 19:NO 20:NO

Salida 2

Comienzo de la ejecución concurrente

1:SI 3:SI 4:SI 5:SI 6:SI 7:SI 8:SI 9:SI 10:SI 11:SI 2:NO 12:NO 13:NO 14:NO 15:NO 16:NO 17:NO 18:NO 19:NO 20:NO

|

Salida 3

Comienzo de la ejecución concurrente

1:SI 3:SI 4:SI 5:SI 6:SI 7:SI 8:SI 9:SI 10:SI 11:SI 2:NO 12:NO 13:NO 14:NO 15:NO 16:NO 17:NO 18:NO 19:NO 20:NO

1 Los atributos de clase son los declarados como static

2 Este atributo de objeto lo definirás como private