

Taller 1b

Santiago Diaz Moreno – 201912247

1.

Si, al ejecutar el programa se imprime 1000 veces 10000 el contador que incrementa 10000. Por lo que el numero que se muestra en consola es 10000000, lo mismo que 1000×10000 .

2.

No, al ejecutar este programa el resultado no va a ser igual al que el del punto uno, esto básicamente porque al crearse un array de mil Threads y al ejecutar el método run que aumentara 100000 veces el contador sobre cada posición de la lista, el numero resultante no va a ser 10000000, sino uno mucho menor.

3.

Ejecución	Valor obtenido
1	7050525
2	9364912
3	9512321
4	8699685
5	8413197

4.

Si, debido a que cada vez que se ejecuta un Thread el metodo run() se corre y la variable compartida se modifica, por ello hay un acceso concurrente.

5.

Ejecución	Valor obtenido	Valor esperado
1	Thread 0 = 91062 Thread 1 = 105041 Thread 2 = 84286	105041
2	Thread 0 = 90390 Thread 1 = 91299 Thread 2 = 94155	94155
3	Thread 0 = 96740 Thread 1 = 68366 Thread 2 = 72351	96740
4	Thread 0 = 92475 Thread 1 = 80282 Thread 2 = 73241	92475
5	Thread 0 = 87361 Thread 1 = 47841 Thread 2 = 59250	87361

6.

Si, debido a que la matriz a pesar de cambiar cada vez que se ejecute el programa, durante la ejecución siempre es la misma y se recorre en el método `run()` cada vez que se ejecuta el Thread de manera concurrente.

7.

Que al utilizar estos tipos de programas nos acortan mucho el código, lo cual beneficia a la memoria y a la velocidad de ejecución, también vimos que las variables globales no cambian por mucho que se ejecuten los threads.