

Taller 1B Mateo Andrés Chaparro Becerra-201730891

Parte1:

1. ¿Al ejecutar el programa, el resultado corresponde al valor esperado?
 - a. Sí, se obtiene el valor de 10,000,000
2. ¿Al ejecutar el programa, el resultado corresponde al valor esperado? Explique.
 - a. No, se obtiene un valor menor a 10,000,000 y cada vez que se corre se obtiene un valor diferente, esto se debe a que los Threads están afectando la misma variable y no están sincronizados, esto causa que se den momentos en que varios Threads intenten acceder a la variable en el mismo momento lo que causa que algunos de ellos no se tomen en cuenta y por tanto no se logre llegar al valor esperado.
3. Ejecute cinco veces el programa y escriba el resultado obtenido en cada ejecución.

Ejecución	Valor Obtenido
1	8089641
2	8513223
3	9772151
4	9954077
5	9905416

4. ¿Hay acceso concurrente a alguna variable compartida? Si es así, diga en dónde
 - a. Sí lo hay, concretamente sobre la variable “contador” y debido a esto es que ocurre el problema observado como se comentó anteriormente.

Parte 2:

1. Ejecute cinco veces el programa y escriba el resultado obtenido en cada ejecución.

Ejecución	Valor Obtenido	Valor esperado
1	64083	103860
2	93544	99051
3	101818	101818
4	103797	103797
5	27342	73951

2. ¿Hay acceso concurrente a alguna variable compartida? Si es así, diga en dónde.
 - a. Sí hay, en este caso se da sobre la variable “mayor” razón por la cual el resultado no siempre es el esperado.
3. ¿Puede obtener alguna conclusión?
 - a. Sí, que es necesario sincronizar apropiadamente los Threads de forma que se pueda aprovechar el paralelismo y el programa funcione adecuadamente, pues de lo contrario en varias ocasiones entregará resultados erróneos como se observó en este laboratorio. En concreto, lo importante es usar la sincronización para evitar el acceso concurrente a variables compartidas.