

- **¿Cuántos años más podremos seguir utilizando esta forma de contar?**

Ese método `currentTimeMillis()` devuelve un entero de tipo `long` (64 bits), y `long` pueden utilizarse para representar valores que van de -9.223.372.372.036.854.775.808 a 9.223.372.036.854.775.807. podemos transformar este valor y obtener el total de días a partir de las milésimas de segundo: 106751 días.

- **¿Qué significa que el tiempo medido sea 0?**

Si el tiempo medio de un algoritmo es 0, significa que se ha ejecutado muy rápido, es decir, desde el punto de vista del tiempo de ejecución es muy rápido.

Sin embargo, no consideramos algoritmos con un tiempo medio inferior a 50.

- **¿Qué pasa con el tiempo si el tamaño del problema se multiplica por 5?**

si multiplico por 5 lo indico como en este bucle `for (int n = 10; n < 100000000; n *= 5)` tendríamos estos resultados para la función suma:

n	t
10	0
50	0
250	0
1250	0
6250	0
31250	0
156250	0
781250	0
3906250	2
19531250	7
97656250	31

por lo tanto, el tiempo máximo es de 31 milisegundos, que no es un valor aceptable.

- **¿Los tiempos obtenidos son los que se esperaban de la complejidad lineal $O(n)$?**

no, porque como hay multiplicación en el bucle for (es decir, $n * 3$), debería ser una complejidad logarítmica.

- **¿Cumplen los valores obtenidos con lo esperado?**

el tiempo se midió en un procesador intel core i7-1185G7, y 16,0 GB de memoria. El algoritmo estuvo funcionando durante más de 10 minutos, y estos son los resultados obtenidos.

n	t_suma	t_maximo
10	31	60
30	67	118
90	151	323
270	508	789
810	1701	1851
2430	5913	5153
7290	15954	13559
21870	48392	43373
65610	143776	134394
196830	436275	402451

