

**Asignatura:** Análisis y Diseño de Algoritmo - Laboratorio 'B'

**Alumno:** Hincho Jove, Angel Eduardo

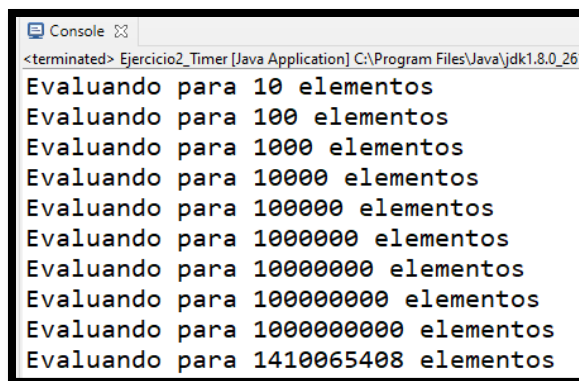
**Semestre:** 2021 - B

Este breve informe complementa lo desarrollado en los Ejercicios 2 y 3, para sus versiones con Timer's o contadores de tiempo incluidos. Veremos como varía el tiempo con respecto a la cantidad de datos mediante gráficas.

### **Ejercicio 2:**

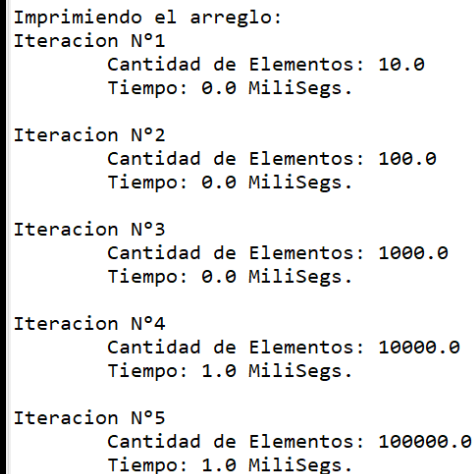
Se nos pide implementar un algoritmo de 'Búsqueda Secuencial', también se pide implementar otros métodos u formas de almacenar los tiempos de ejecución o demora para cierta cantidad de elementos. Esto se logró guardando la cantidad de datos y sus tiempos respectivos en un arreglo bidimensional de dos columnas, la primera columna almacena la cantidad de elementos con la que se trabajo mientras que la siguiente columna almacena el tiempo de ejecución.

La primera salida por consola es una impresión de los datos que fueron evaluados.



```
<terminated> Ejercicio2_Timer [Java Application] C:\Program Files\Java\jdk1.8.0_261
Evaluando para 10 elementos
Evaluando para 100 elementos
Evaluando para 1000 elementos
Evaluando para 10000 elementos
Evaluando para 100000 elementos
Evaluando para 1000000 elementos
Evaluando para 10000000 elementos
Evaluando para 100000000 elementos
Evaluando para 1000000000 elementos
Evaluando para 1410065408 elementos
```

Luego se imprime según las iteraciones que se realizaron, tomar en cuenta que la cantidad de datos ingresados son potencias de 10, empezando con 10, 100, 1000 y así sucesivamente. Veamos la impresión de los tiempos requeridos para cada iteración con su respectiva cantidad de datos.



```
Imprimiendo el arreglo:
Iteracion N°1
  Cantidad de Elementos: 10.0
  Tiempo: 0.0 MiliSegs.

Iteracion N°2
  Cantidad de Elementos: 100.0
  Tiempo: 0.0 MiliSegs.

Iteracion N°3
  Cantidad de Elementos: 1000.0
  Tiempo: 0.0 MiliSegs.

Iteracion N°4
  Cantidad de Elementos: 10000.0
  Tiempo: 1.0 MiliSegs.

Iteracion N°5
  Cantidad de Elementos: 100000.0
  Tiempo: 1.0 MiliSegs.
```

```

Iteracion N°6
  Cantidad de Elementos: 1000000.0
  Tiempo: 2.0 MiliSegs.

Iteracion N°7
  Cantidad de Elementos: 1.0E7
  Tiempo: 5.0 MiliSegs.

Iteracion N°8
  Cantidad de Elementos: 1.0E8
  Tiempo: 37.0 MiliSegs.

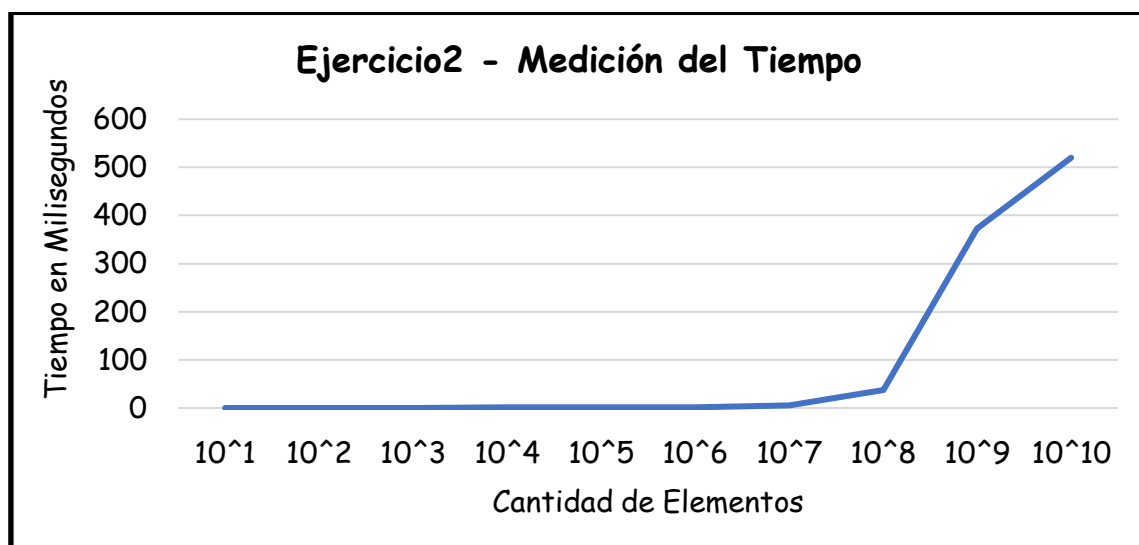
Iteracion N°9
  Cantidad de Elementos: 1.0E9
  Tiempo: 373.0 MiliSegs.

Iteracion N°10
  Cantidad de Elementos: 1.410065408E9
  Tiempo: 520.0 MiliSegs.

```

Con ayuda de una Hoja de Cálculo se obtuvo la siguiente gráfica que contrasta la cantidad de datos contra el tiempo que demoro su procesamiento.

Cantidad de Elementos	Tiempo en Milisegundos
$10^1$	0
$10^2$	0
$10^3$	0
$10^4$	1
$10^5$	1
$10^6$	2
$10^7$	5
$10^8$	37
$10^9$	373
$10^{10}$	520



### Ejercicio 3:

Al igual que el Ejercicio 2, se pide ordenar un arreglo estándar. En este caso también será un arreglo estándar de enteros los cuales serán ordenados usando el algoritmo de 'Ordenación por Inserción'. De igual manera se guardarán las cantidades de datos y tiempos respectivos de ejecución en un arreglo bidimensional siendo la primera columna la 'Cantidad de Datos' y la segunda el 'Tiempo en Milisegundos' para cada iteración.

La primera salida por consola son los valores que se evaluaron, así como los tiempos respectivos para cada iteración y sus cantidades de elementos.

```
Evaluando para 10 elementos  
Evaluando para 100 elementos  
Evaluando para 1000 elementos  
Evaluando para 10000 elementos  
Evaluando para 100000 elementos  
Evaluando para 1000000 elementos
```

La otra salida por consola son los datos por cada iteración, datos como 'Cantidad de elementos' y 'Tiempo de ejecución'. De igual manera, con ayuda de una Hoja de Cálculo, se armará el gráfico respectivo para el Ejercicio 3

```
Imprimiendo el arreglo:  
Iteracion N°1  
    Cantidad de Elementos: 10.0  
    Tiempo: 0.0 MiliSegs.  
  
Iteracion N°2  
    Cantidad de Elementos: 100.0  
    Tiempo: 0.0 MiliSegs.  
  
Iteracion N°3  
    Cantidad de Elementos: 1000.0  
    Tiempo: 3.0 MiliSegs.  
  
Iteracion N°4  
    Cantidad de Elementos: 10000.0  
    Tiempo: 40.0 MiliSegs.  
  
Iteracion N°5  
    Cantidad de Elementos: 100000.0  
    Tiempo: 1993.0 MiliSegs.  
  
Iteracion N°6  
    Cantidad de Elementos: 1000000.0  
    Tiempo: 213636.0 MiliSegs.
```

Cantidad de Elementos	Tiempo en Milisegundos
$10^1$	0
$10^2$	0
$10^3$	3
$10^4$	4
$10^5$	1993
$10^6$	213636

