

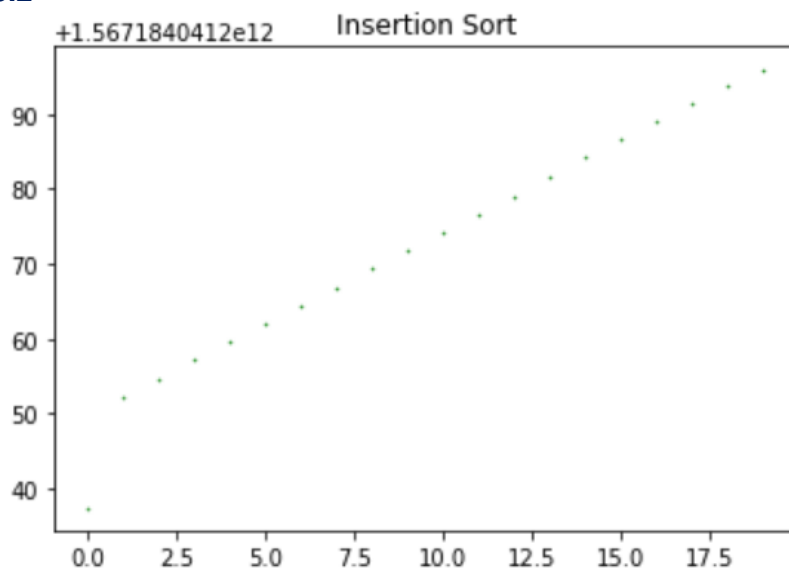
Laboratorio Nro. 02 Notación O grande

Camilo Cossio Alzate
Universidad Eafit
Medellín, Colombia
ccossioa@eafit.edu.co

Juan Andrés Giraldo Aristizábal
Universidad Eafit
Medellín, Colombia
jagiraldoa@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

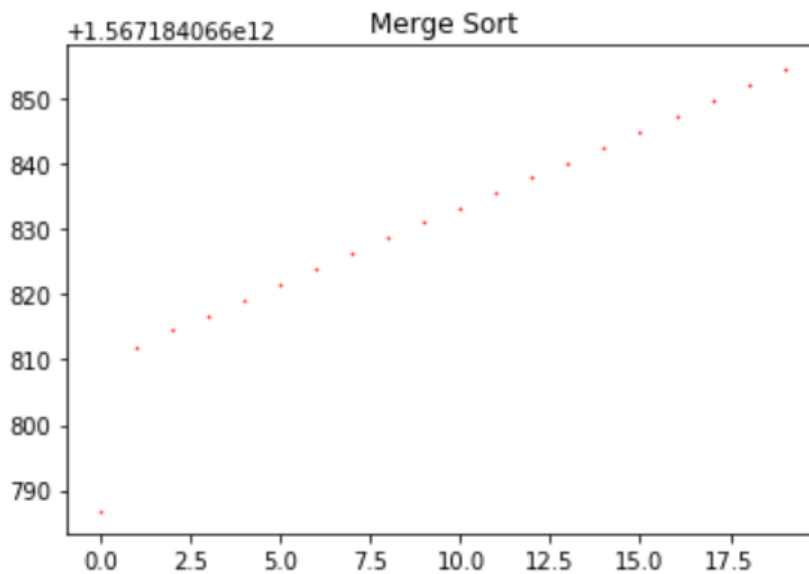
3.1
3.2



PhD. Mauricio Toro Bermúdez
Docente | Escuela de Ingeniería | Informática y Sistemas
Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1

Código ST0245



3.3 Según lo que podemos apreciar de las gráficas, es mucho más eficiente Merge Sort con arreglos de gran tamaño.

3.4 No es apropiado porque tardaría mucho tiempo, sería más apropiado usar Merge Sort.

3.5 Para que Insertion Sort sea más efectivo que Merge Sort en arreglos de gran tamaño, es necesario que los datos del arreglo sean muy pequeños.

3.6

```
public int maxSpan(int[] nums) {
    if (nums.length == 0) return 0;
    if (nums.length == 1) return 1;
    int span = 0;
    for (int i = 0; i < nums.length; i++){
        for (int j = nums.length-1; j >= 0; j--){
            if (nums[i]==nums[j]){
                if ((j-i)>span) span = j-i;
            }
        }
    }
    return span+1;
}
```

El algoritmo funciona de la siguiente manera:

Primero, se verifica que el tamaño del arreglo sea mayor a uno, si el arreglo no tiene posiciones o tiene solo una, se retorna 0 y 1 respectivamente porque este sería el span máximo. Luego se inicializa la variable span en 0, y se hacen dos ciclos anidados, el de afuera recorre el arreglo del inicio al final. El ciclo anidado recorre el arreglo del final hasta el inicio, lo que permite comparar una posición del arreglo con el resto. Dentro de los dos ciclos se encuentra un condicional que determina si un número en posición i es igual a otro en posición j; si los números son iguales entra a otro condicional que se encarga de mirar cuantas posiciones de diferencia existen entre dichos números, esto es a lo que llamamos span, si dicha diferencia es mayor a lo que se encuentra en ese momento en la variable span, esta pasa a ser la diferencia entre i y j, sino span no cambia, y se sigue ejecutando el

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
 Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
 Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1

Código ST0245

ciclo normalmente. Cuando se termina con el proceso de los ciclos, finalmente se retorna la variable $\text{span}+1$, esto porque cuando se resta $i-j$, siempre da como resultado un número que es una unidad menor que el verdadero span. Por ejemplo, si en la ejecución del algoritmo se encuentra que el número en la posición $i=0$ es igual que el número en la posición $j=4$, restando $i-j$ nos daría como resultado 3, pero el verdadero span sería 4.

3.7

| Nombre del ejercicio | Complejidad |
|----------------------|-------------|
| Sum28 | $O(n)$ |
| More14 | $O(n)$ |
| fizzArray | $O(n)$ |
| Only14 | $O(n)$ |
| fizzArray2 | $O(n)$ |

3.8

| Nombre del ejercicio | ¿Qué es n? |
|----------------------|-----------------------------------|
| Sum28 | $n = \text{Longitud del arreglo}$ |
| More14 | $n = \text{Longitud del arreglo}$ |
| fizzArray | $n = \text{Longitud del arreglo}$ |
| Only14 | $n = \text{Longitud del arreglo}$ |
| fizzArray2 | $n = \text{Longitud del arreglo}$ |

4) Simulacro de Parcial

4.1 c

4.2 b

4.3 b

4.4 b

4.5 d

4.6 a

4.7.1 $T(n)=T(n-1)+C$

4.7.2 $O(n)$

4.8 a

4.9 d

4.11 c

4.12 b

4.13 b

4.14 a

5) Lectura recomendada (opcional)

Mapa conceptual

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
 Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
 Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1
Código ST0245

6) Trabajo en Equipo y Progreso Gradual (Opcional)

6.1 *Actas de reunión*

6.2 *El reporte de cambios en el código*

6.3 *El reporte de cambios del informe de laboratorio*

Referencias:

Kumra, M (sin fecha) Python program for implementation of Insertion Sort (Version 1.0) [Sort code]. <https://www.geeksforgeeks.org/insertion-sort/>

Khanna, M (sin fecha) Python program for implementation of MergeSort (Version 1.0) [Sort code]. <https://www.geeksforgeeks.org/merge-sort/>