

## Laboratorio Nro. 2

### Complejidad de algoritmos

**David Ruiz Echeverri**  
Universidad Eafit  
Medellín, Colombia  
druize@eafit.edu.co

**Tomás Gaviria Ossa**  
Universidad Eafit  
Medellín, Colombia  
tgaviriao@eafit.edu.co

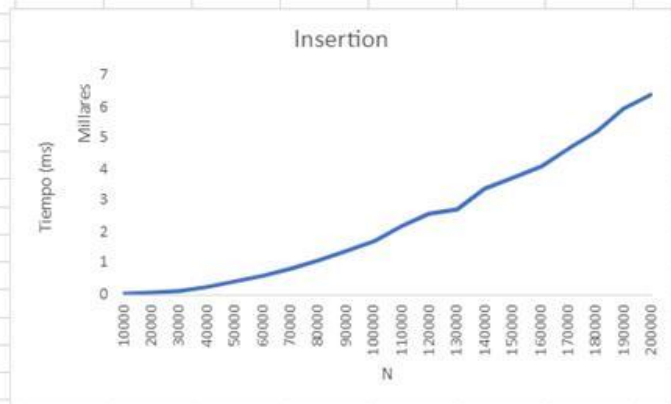
### 3) Simulacro de preguntas de sustentación de Proyectos

#### 1.1/3.1/3.2:

```

Opciones
el tiempo de insertionSort es de: 22471
el tiempo de mergeSort es de: 49
  
```

N	Tiempo
10000	23
20000	70
30000	146
40000	259
50000	434
60000	610
70000	831
80000	1116
90000	1435
100000	1712
110000	2229
120000	2626
130000	2772
140000	3407
150000	3783
160000	4151
170000	4694
180000	5245
190000	6018
200000	6451



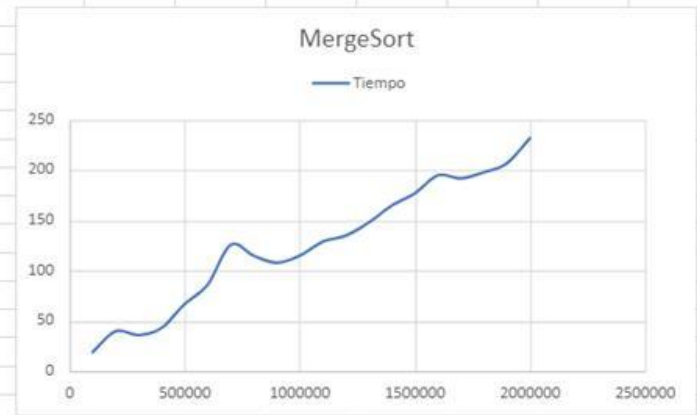
**PhD. Mauricio Toro Bermúdez**

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

N	Tiempo
100000	19
200000	40
300000	36
400000	43
500000	67
600000	86
700000	126
800000	115
900000	108
1000000	115
1100000	129
1200000	135
1300000	148
1400000	165
1500000	177
1600000	195
1700000	192
1800000	198
1900000	207
2000000	232



**3.3** No, no es recomendado usar insertion sort debido a su complejidad ( $O(n^2)$ ). Esto significa a que cuando introducimos elementos  $n$  al algoritmo este tardaría mucho tiempo en procesarlos, no cumpliría con la demanda esperada.

**3.4** Esto se debe a que en el algoritmos de merge sort encontramos una división repetida a la hora de trabajar

### 3.7/3.8

#### bigDiff:

- ❖ Ecuación de recurrencia:  $T(n) = c*(n-1)*n = cn^2 - cn$
- ❖ Gran O :  $O(n^2)$
- ❖  $n$  = largo del arreglo

#### canBalance:

- ❖ Ecuación de recurrencia:  $T(n) = c*n^2$
- ❖ Gran O :  $O(n^2)$
- ❖  $n$  = largo del arreglo

#### centeredAverage:

- ❖ Ecuación de recurrencia:  $T(n) = c*(n-1)*n = cn^2 - cn$
- ❖ Gran O :  $O(n^2)$
- ❖  $n$  = largo del arreglo

#### countEvents:

- ❖ Ecuación de recurrencia:  $T(n) = c*n$
- ❖ Gran O :  $O(n)$
- ❖  $n$  = largo del arreglo

#### fix34:

**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

- ❖ Ecuación de recurrencia:  $T(n) = c \cdot n^2$
- ❖ Gran O :  $O(n^2)$
- ❖  $n$  = largo del arreglo

#### fix45:

- ❖ Ecuación de recurrencia:  $T(n) = c \cdot n^2$
- ❖ Gran O :  $O(n^2)$
- ❖  $n$  = largo del arreglo

#### haveThree:

- ❖ Ecuación de recurrencia:  $T(n) = c \cdot n$
- ❖ Gran O :  $O(n)$
- ❖  $n$  = largo del arreglo

#### linearIn:

- ❖ Ecuación de recurrencia:  $T(n) = c \cdot n$
- ❖ Gran O :  $O(n)$
- ❖  $n$  = largo del arreglo externo (mas largo)

#### maxSpan:

- ❖ Ecuación de recurrencia:  $T(n) = c \cdot n$
- ❖ Gran O :  $O(n)$
- ❖  $n$  = largo del arreglo

#### sum13:

- ❖ Ecuación de recurrencia:  $T(n) = c \cdot n$
- ❖ Gran O :  $O(n)$
- ❖  $n$  = largo del arreglo

### 4) Simulacro de Parcial

**4.1** El algoritmo tardaría 100ms.

**4.2** D.

**4.3** A.

**4.4**

4.4.1  $O(n \cdot m)$

4.4.2  $O(n \cdot m)$

**4.5**

4.5.1 D.

4.5.2 A.

**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