

Laboratorio Nro. 2

Notación O grande

Juan Diego Acevedo Correa
Universidad Eafit
Medellín, Colombia
jdacevedoc@eafit.edu.co

José Alejandro Sánchez Sánchez
Universidad Eafit
Medellín, Colombia
jasanchez@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

3.1

	N = 40000	N = 60000	N = 80000	N = 100000
Insertion Sort	310	715	1246	1934

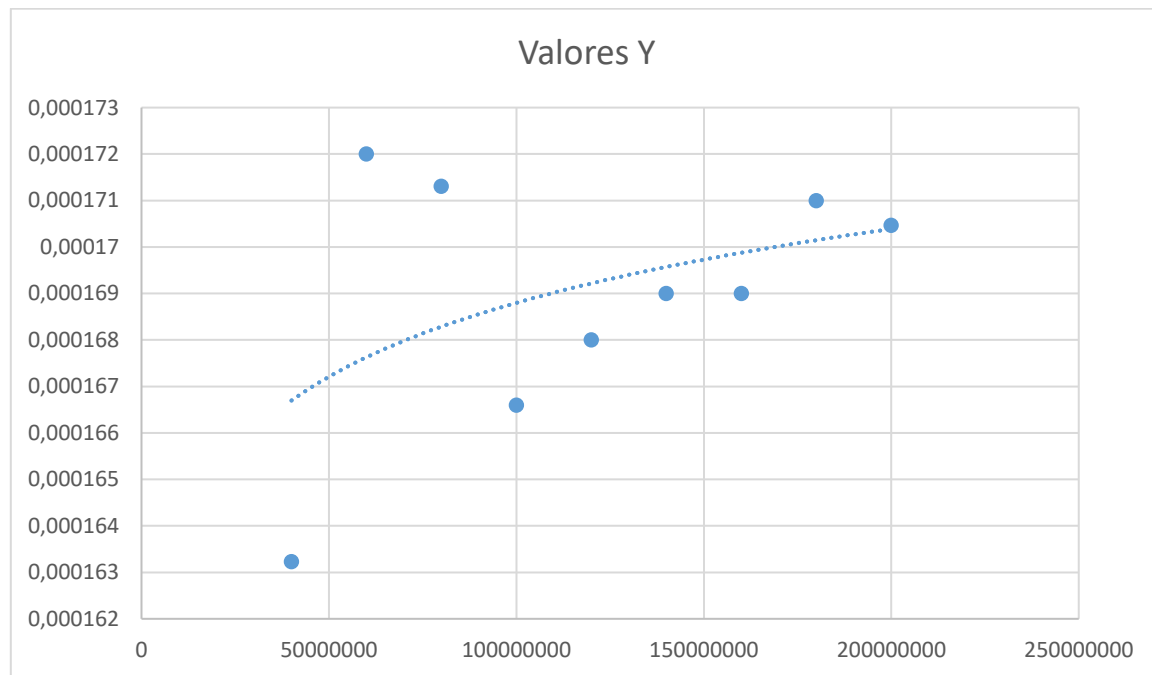
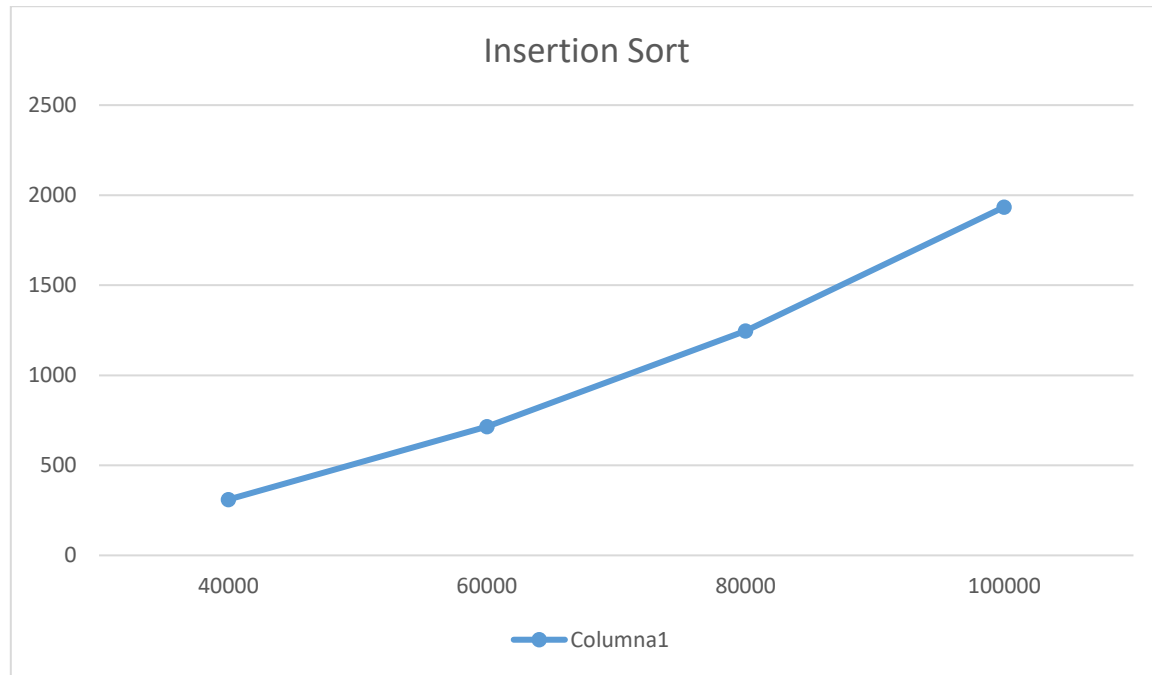
<i>Merge Sort</i>	
Entrada N	Milisegundos
40000000	0.00016323
60000000	1.72E-04
80000000	0.00017131
100000000	0.00016660
120000000	1.68E-04
140000000	1.69E-04
160000000	1.69E-04
180000000	1.71E-04
200000000	0.00017047

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** Merge Sort nos da buenos tiempos a la hora de procesar entradas grandes , mientras que Insertion Sort nos encontraríamos un tiempo de ejecución mayor y un pero rendimiento en ciertas ocasiones.

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

- 3.4** No sería apropiado utilizar Insertion Sort en un videojuego con millones de elementos ya que por las pruebas que realizamos anteriormente pudimos ver que no se conforma de la mejor forma a la hora de ejecutar una gran cantidad de elementos y nos presentaría un tiempo demasiado grande de ejecución.
- 3.5** Para que Insertion Sort sea eficiente el problema al que se enfrenta no debe exceder una complejidad de máximo $O(n^2)$.
- 3.6**

4) Simulacro de Parcial

- 4.1** C
- 4.2** B
- 4.3** B