

Laboratorio Nro. 2 Complejidad

Luis Fernando Vargas Agudelo
Universidad Eafit
Medellín, Colombia
lvarga12@eafit.edu.co

Tomás Bedoya Henao
Universidad Eafit
Medellín, Colombia
tbedoyah@eafit.edu.co

1) Tiempos de ejecución de los algoritmos Insertion Sort y Merge Sort

Para medir los tiempos de ejecución de los diferentes algoritmos se utilizaron tres librerías las cuales ajustando los parámetros de máxima capacidad de recursión que el intérprete en este caso Python puede tener, la longitud del arreglo con el cual se ejecuta el algoritmo, creación de variables aleatorias dentro del arreglo, esto con el fin de que cada arreglo fuese totalmente diferente al anterior y así obtener datos más precisos. Adicional a esto se utilizó la librería time con el fin de conocer los milisegundos que tomaba cada algoritmo en realizar su ejecución.

A continuación, se encuentra la complejidad asintótica, notación O, para cada uno de los algoritmos y los tiempos de ejecución en milisegundos para 20 casos diferentes.

<i>Algoritmo</i>	Insertion Sort	Merge Sort
<i>Complejidad</i>	$O(n^2)$	$O(n * \log n)$

Tamaño del Problema	<i>Tiempo (milisegundos)</i>	<i>Tiempo (milisegundos)</i>
500	25.92	3.97
1000	109.94	7.97
1500	229.87	11.96
2000	544.83	15.95
2500	686.93	21.94
3000	984.36	26.92
3500	1351.42	30.91
4000	1862.02	37.93
4500	2190.79	41.84
5000	3328.13	45.88
5500	3336.06	50.86
6000	3930.10	56.84
6500	4466.94	61.87
7000	5251.48	68.82
7500	6112.83	70.80
8000	7080.69	76.79

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

8500	7831.30	85.73
9000	8945.41	88.80
10000	10629.02	94.74
10500	11888.86	100.69

3) Simulacro de preguntas de sustentación de Proyectos

3.1

<i>Algoritmo</i>	Insertion Sort	Merge Sort
<i>Complejidad</i>	$O(n^2)$	$O(n * \log n)$

Tamaño del Problema	<i>Tiempo (milisegundos)</i>	<i>Tiempo (milisegundos)</i>
500	25.92	3.97
1000	109.94	7.97
1500	229.87	11.96
2000	544.83	15.95
2500	686.93	21.94
3000	984.36	26.92
3500	1351.42	30.91
4000	1862.02	37.93
4500	2190.79	41.84
5000	3328.13	45.88
5500	3336.06	50.86
6000	3930.10	56.84
6500	4466.94	61.87
7000	5251.48	68.82
7500	6112.83	70.80
8000	7080.69	76.79
8500	7831.30	85.73
9000	8945.41	88.80
10000	10629.02	94.74
10500	11888.86	100.69

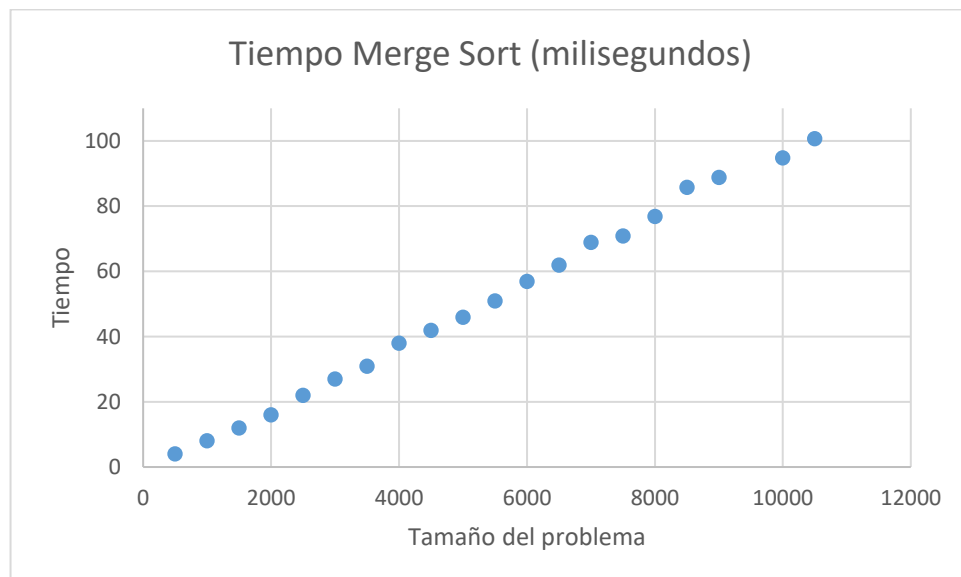
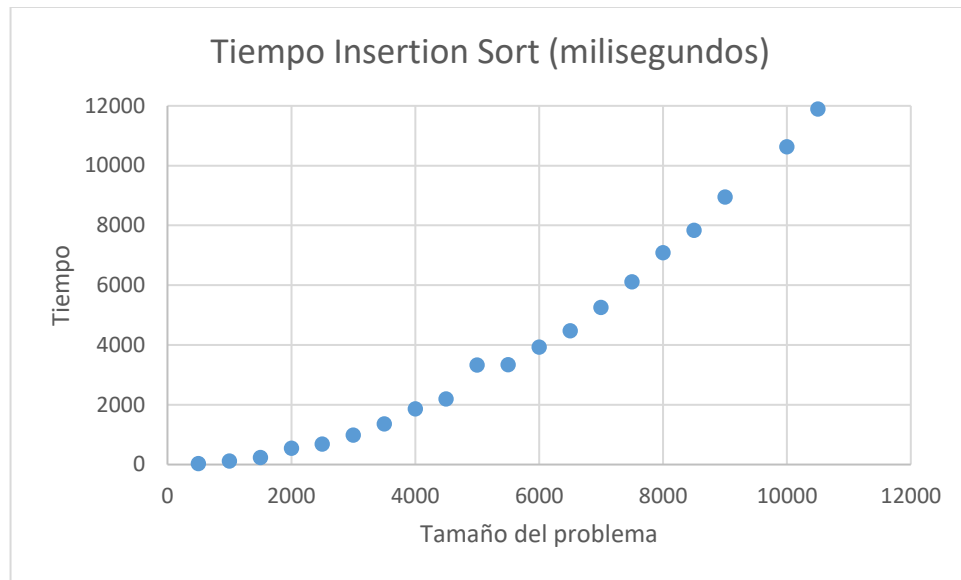
PhD. Mauricio Toro Bermúdez

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

3.2



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

Luego de analizar ambos algoritmos con diferentes tamaños de problema es posible afirmar que la diferencia en la complejidad de los mismo es notoria, la eficiencia que presenta Merge Sort para arreglos grandes es mucho más alta que la brindada por Insertion Sort, para entender un poco más cómo se comportan con tamaños de problemas grandes se realizará una relación de tiempo entre ellos, para una solución con un arreglo de 10000 números Insertion Sort se tomó un tiempo de 10,62 segundos mientras que Merge Sort se demoró para el mismo tamaño 94,74 milisegundos, es decir que Merge Sort soluciona el mismo problema 112,19 veces más rápido que Insertion Sort, esto se ve aún más marcado en problemas con un tamaño más grande.

Debido a esto, utilizar el algoritmo Insertion Sort para problemas con un tamaño de millones de números no es viable y tardaría muchísimo en comparación con Merge Sort. Precisamente por esto Merge Sort es el algoritmo indicado para obtener resultados de arreglos grandes en tiempo real, como pueden ser los renderizados necesarios en juegos como Silent Hill, debido a que es indispensable brindarle la fluidez de juego pertinente a los usuarios y evitar grandes tiempos de carga.

3.4

3.5

3.6

4) Simulacro de Parcial

4.1 a

4.2 b

4.3 length-1

5) Lectura recomendada (opcional)

Mapa conceptual

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

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