

Algoritmos de Ordenamiento

Estructuras de Datos Avanzadas

Mariana Zapata Covarrubias 195111
A 20 de feb. de 23

Índice

Objetivo.....	3
<i>Análisis tiempo de ordenamiento</i>	3
Datos Ordenados.....	3
Análisis	3
Gráfica	3
Datos Ordenados Inversamente.....	4
Análisis	4
Gráfica	4
Datos Ordenados Aleatoriamente.....	5
Análisis	5
Gráfica	5
<i>Análisis número de comparaciones.....</i>	6
Datos Ordenados.....	6
Análisis	6
Gráfica	6
Datos Ordenados Inversamente.....	7
Análisis	7
Gráfica	7
Datos Ordenados Aleatoriamente.....	8
Análisis	8
Gráfica	8

Objetivo

El objetivo de esta tarea es determinar empíricamente el desempeño de los algoritmos de ordenamiento vistos en clase.

Análisis tiempo de ordenamiento

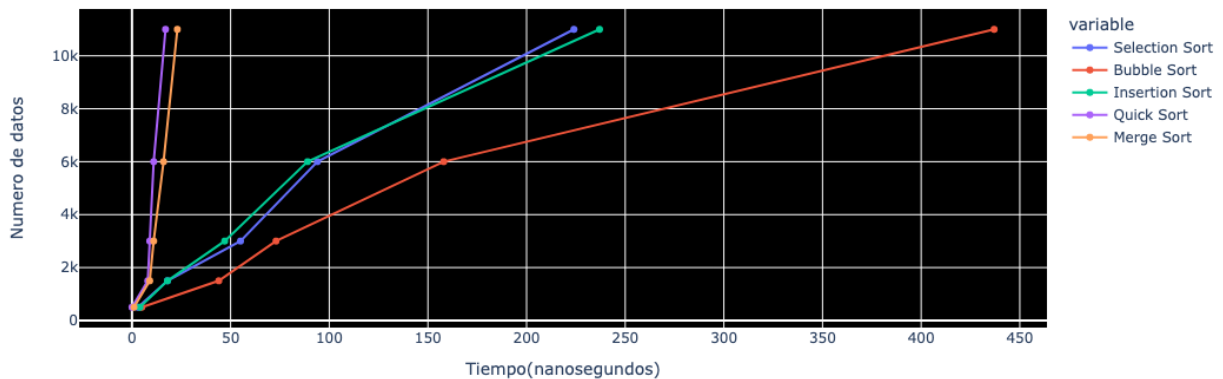
Datos Ordenados

Análisis

- A plena vista podemos observar que los métodos Merge Sort y Quick Sort tienen un comportamiento más eficaz al momento de ordenar.
- El método Bubble Sort es el método que más varía conforme a la cantidad de datos que se le piden ordenar.
- En un punto medio de desempeño, tenemos a los métodos Selection Sort e Insertion Sort. Notemos que, bajo el contexto de un arreglo ordenado, su proceso de ordenamiento se comporta de manera similar.

Gráfica

Desempeño de los algoritmos de ordenamiento con un arreglo ordenado



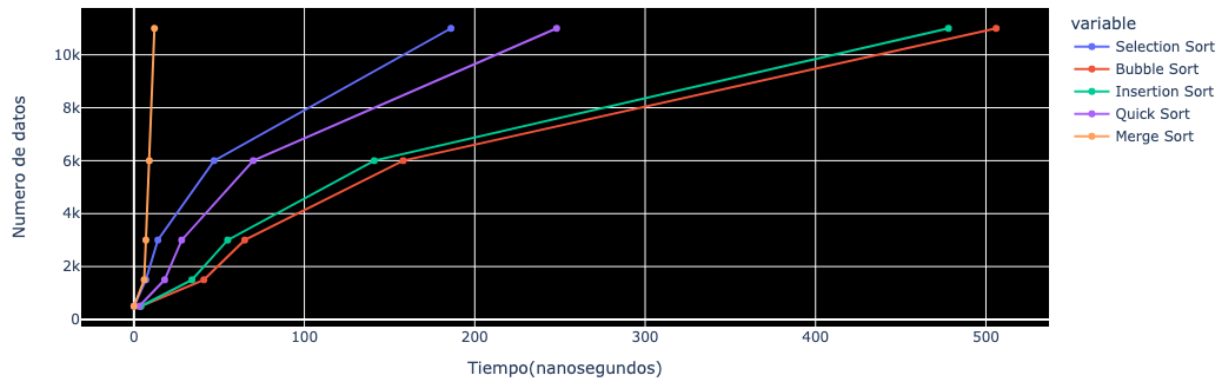
Datos Ordenados Inversamente

Análisis

- Los métodos Quick Sort y Merge Sort dejan de ser sencillos en cuanto a tiempo de ordenamiento. Vemos, pues, que la característica que habíamos encontrado anteriormente cambia. Ahora el Merge Sort sigue una tendencia de bajo tiempo de ordenamiento de datos, mientras que el Quick Sort se diverge de esta tendencia que había tenido anteriormente.
- Por otro lado, algo que cabe mencionar es que el Selection Sort aquí pareciera ser más eficaz. Es decir, bajo un contexto de ordenar un arreglo que esta ordenado inversamente.
- Un rasgo distintivo y permanente en el ordenamiento del Bubble Sort es que es el más lento de los 5 métodos de ordenamiento.

Gráfica

Desempeño de los algoritmos de ordenamiento con un arreglo ordenado inversamente



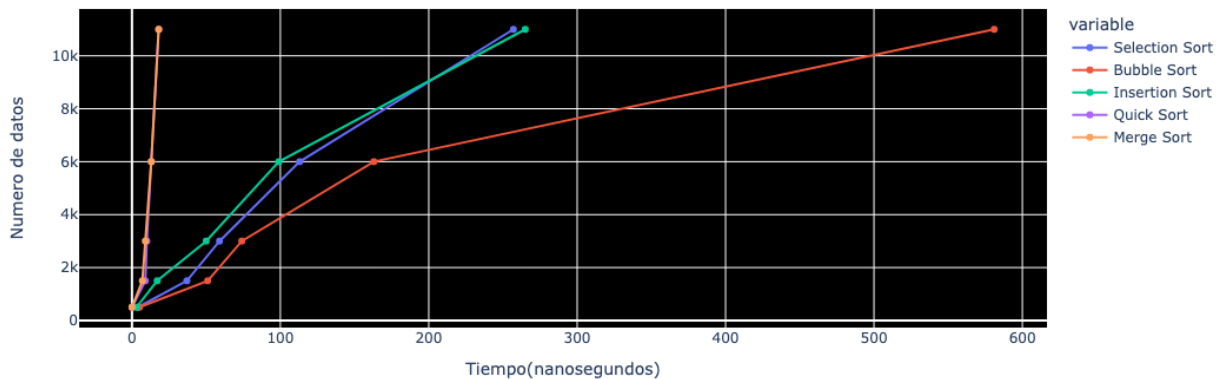
Datos Ordenados Aleatoriamente

Análisis

- Tanto Merge Sort como Quick Sort siguen una tendencia similar de desempeño de ordenamiento. Vemos que tienen exactamente el mismo tiempo de ordenamiento con un arreglo de 11,000 datos. Difiriendo un poco con arreglos de menos de dimensiones de 500, 1,500 y 3,000 datos, siendo Quick Sort más rápido y convergiendo en tiempo de ejecución en un arreglo de 11,000 datos.
- Volvemos a observar que Insertion Sort y Selection Sort vuelven a verse inmersos en una dinámica de centinelas, en cuanto a tiempo de desempeño se refiere. Observamos que el Selection Sort pareciera más rápido en el ordenamiento de arrays menores a 11,000 datos.
- El Bubble Sort mantiene la tendencia, en general, de ser el algoritmo de ordenamiento más lento. Creciendo en tiempo de manera notable mientras crece, a su vez, la cantidad de datos a ordenar.

Gráfica

Desempeño de los algoritmos de ordenamiento con un arreglo ordenado aleatoriamente



Análisis número de comparaciones

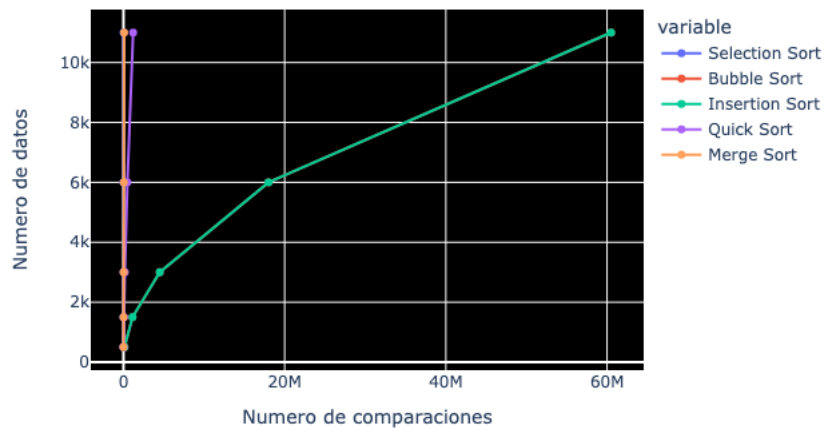
Datos Ordenados

Análisis

- El Selection Sort realiza el mismo número de comparaciones que de datos menos uno. Siendo, pues, el método de ordenamiento que compara menos en su camino a ordenar.
- Siguiendo esta tendencia en cuanto a cantidad de comparaciones está el Merge Sort que hace, exactamente, la misma cantidad de comparaciones que de datos dados.
- En temas de cantidad de comparaciones el que encabeza el tercer lugar es el método Quick Sort.
- El Bubble Sort e Insertion Sort siguen, exactamente, la misma tendencia de crecimiento en cuanto comparaciones se refiere.

Gráfica

Desempeño de los algoritmos de ordenamiento con un arreglo ordenado



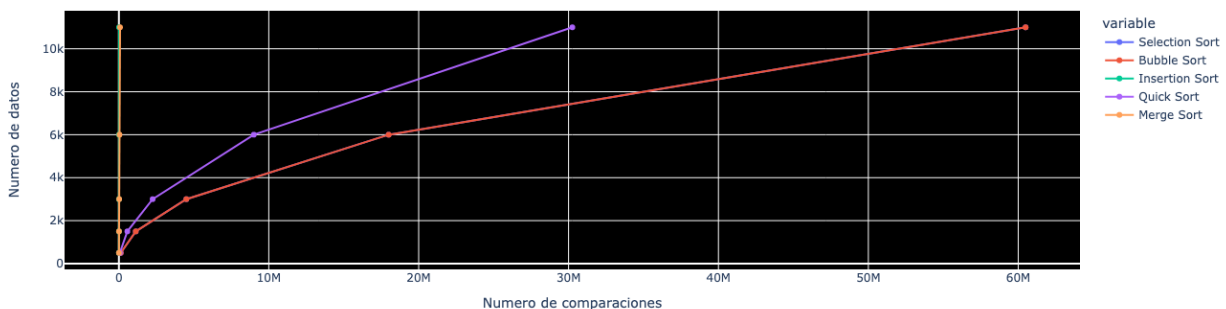
Datos Ordenados Inversamente

Análisis

- El Merge Sort encabeza la lista de menos comparaciones realizadas junto con Insertion Sort. Si bien es cierto que sus números, en el caso de Merge Sort, se disparan de manera considerable en comparación con el ejercicio anterior, también es cierto que este método diverge mucho su comportamiento a comparación de los otros métodos, con excepción del Insertion Sort.
- Tanto Selection Sort como Quick Sort brincan a las decenas de millones de comparaciones. Situación que no se había visto en el ejercicio anterior del análisis. Siendo el Selection Sort el que crece en mayor cantidad de comparaciones llegando hasta la cantidad de casi 60.5 millones de comparaciones para el ordenamiento de 11,000 datos. El Quick Sort, en su extremo más alto llega la cantidad de 30.25 millones de comparaciones para el ordenamiento de dicha cantidad.
- El Bubble Sort mantiene una tendencia de cantidad de comparaciones en decenas de millones. Siendo el método que presenta más cantidad de comparaciones hasta el momento.

Gráfica

Desempeño de los algoritmos de ordenamiento con un arreglo ordenado inversamente



Datos Ordenados Aleatoriamente

Análisis

- Los métodos Merge Sort y Quick Sort siguen una dinámica de centinelas en cuanto a cantidad de comparaciones se refiere. Su cantidad de comparaciones no dista mucho y siguen una dinámica de crecimiento similar, bajo sus respectivas cantidades de datos.
- El método Insertion Sort mantiene una dinámica de crecimiento en cantidad de comparaciones alta, en comparación con Quick Sort y Merge Sort.
- El Bubble Sort presenta una tendencia alta en cuanto a cantidad de comparaciones en el proceso de ordenamiento. El Selection Sort se le une en la misma cantidad y tendencia de comparaciones. Este par de métodos son los que tienen una mayor cantidad de comparaciones bajo este contexto de análisis.

Gráfica

Desempeño de los algoritmos de ordenamiento con un arreglo ordenado aleatoriamente

