

Comparativa Árboles

Salvador Romero Cortés

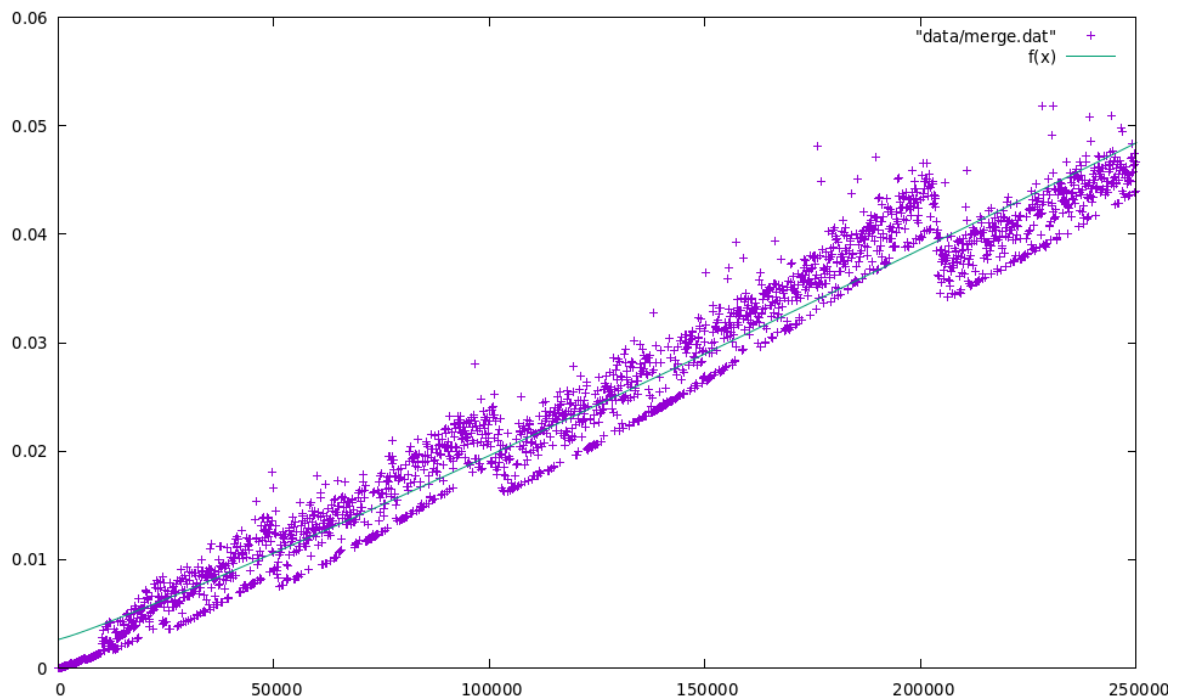
En esta práctica comparamos los tiempos de ejecución de los distintos métodos de ordenación:

- Mergesort
- Heapsort
- Construcción de un AVL

Mergesort

Para probar este algoritmo usamos el archivo `mergesort.cpp` incluido en los archivos fuente de la práctica 1.

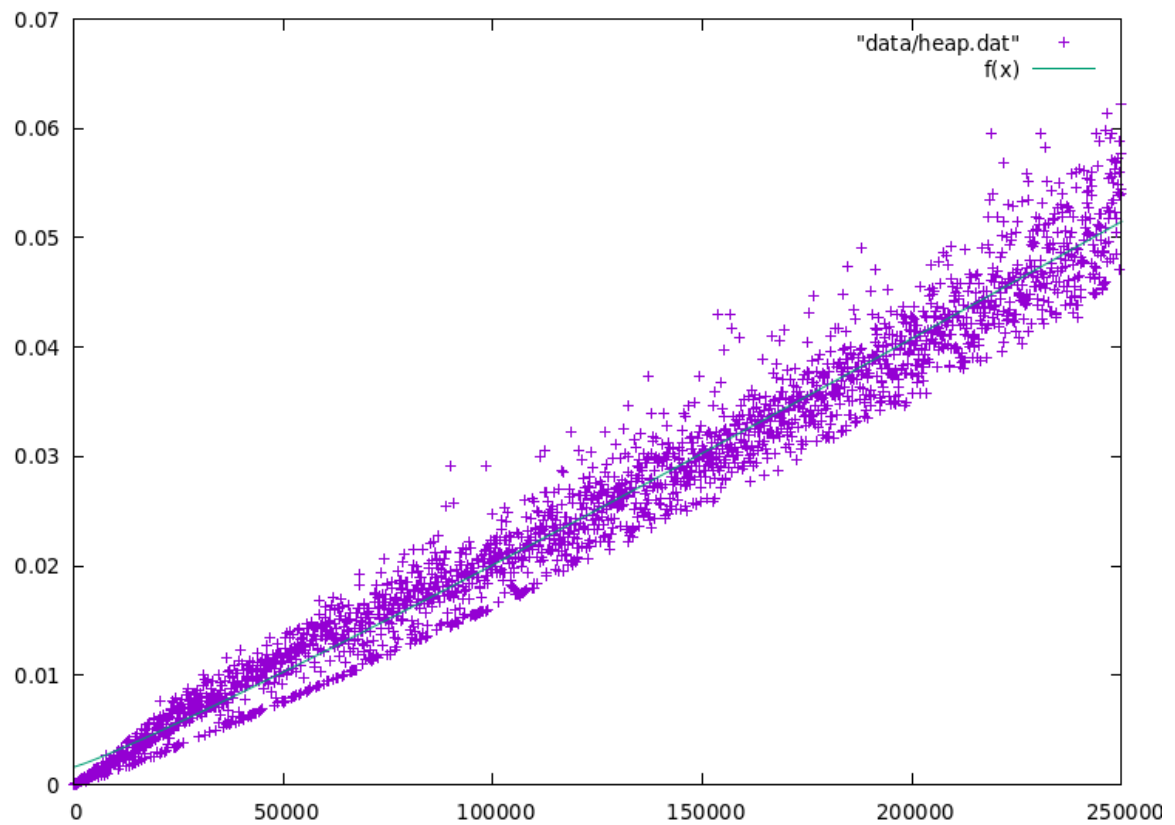
Tras ejecutar el programa con hasta 250000 elementos obtenemos el siguiente resultado.



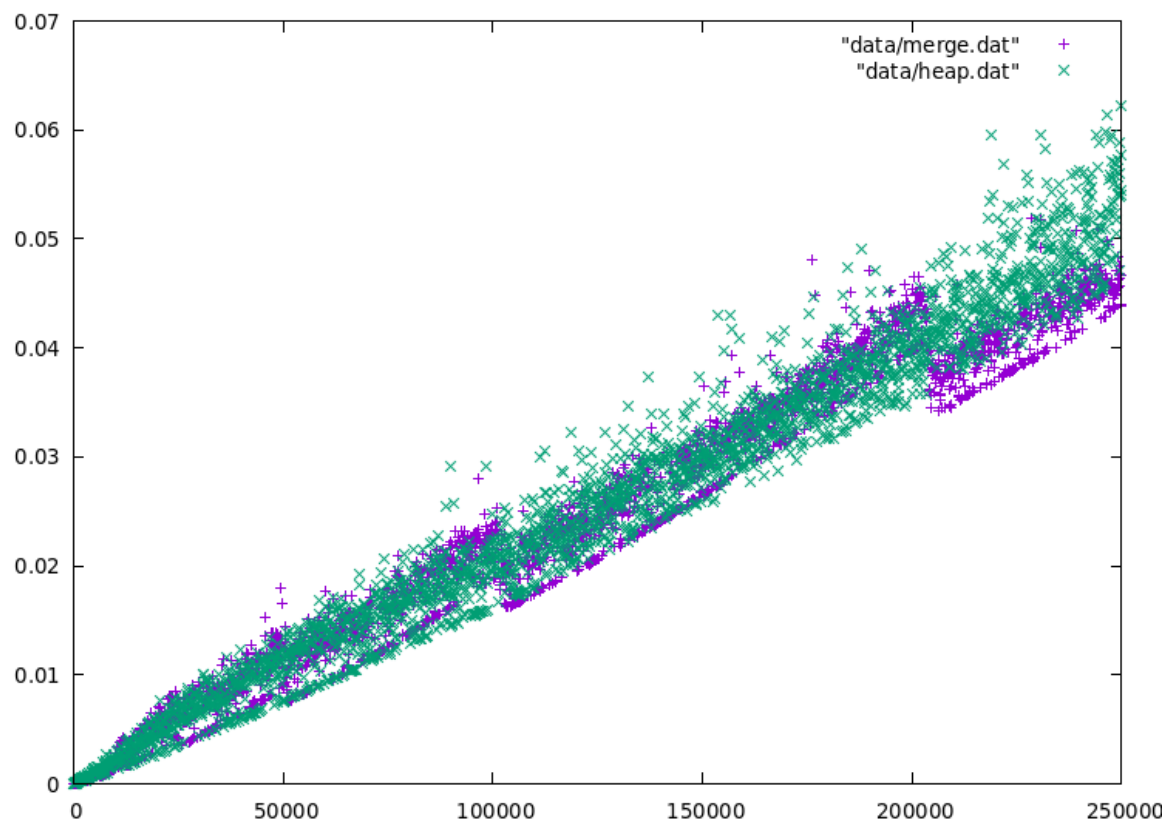
Vemos que la gráfica tiene una forma similar a $n \log(n)$.

Heapsort

Para usar heapsort, ejecutamos un programa que inserte y elimine elementos de un árbol APO.



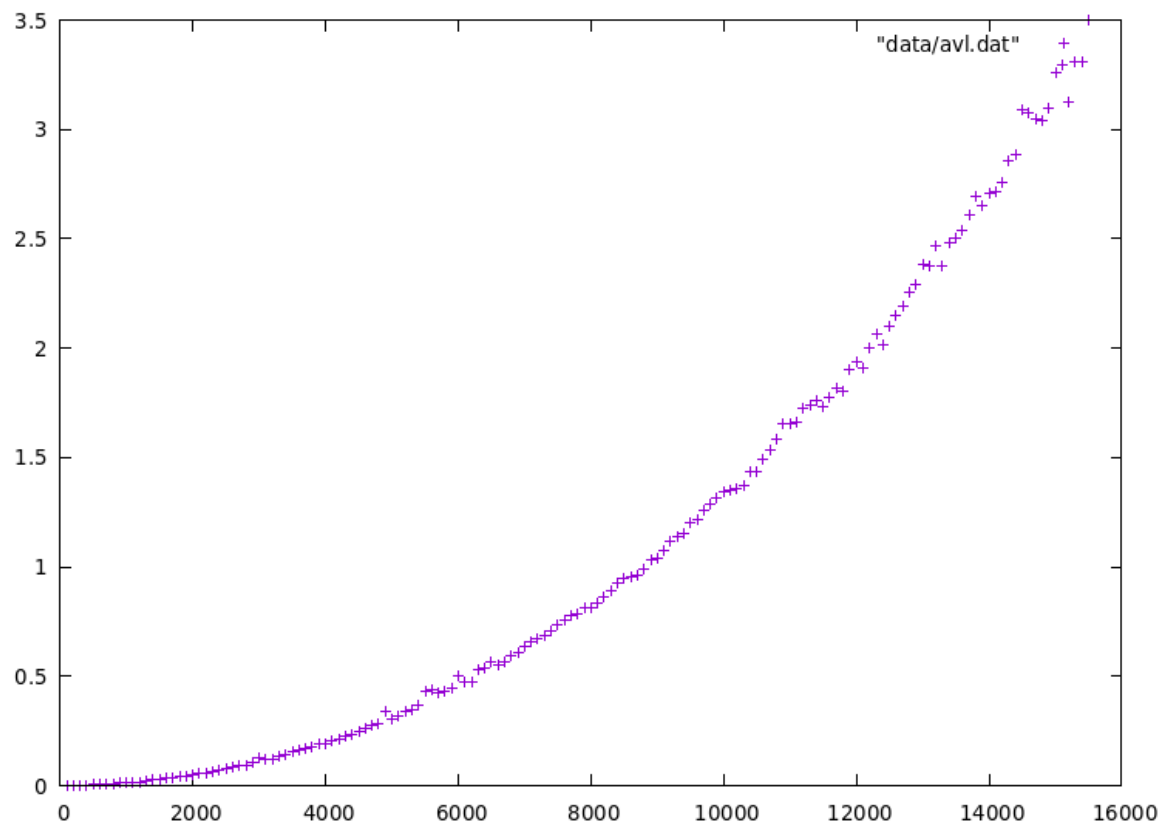
Vemos que los resultados son increíblemente similares a los de mergesort. De hecho si superponemos los resultados:



Vemos que se superponen durante casi todo el tiempo.

AVL

En este caso insertamos elementos en un árbol AVL que inserta ordenados los elementos. Teóricamente debería tener una complejidad de $O(\log(n))$ en el peor caso, sin embargo, en la práctica obtenemos un resultado con complejidad cuadrática ($O(n^2)$), algo bastante alejado de lo esperado.



Conclusiones

Por tanto, a vista de los resultados obtenidos, a la hora de ordenar elementos podemos usar heapsort o mergesort para obtener resultados similares. Si el árbol AVL, funcionase correctamente sería la mejor opción, pero como no es así queda como la peor debido a su pésimo rendimiento.