Estructuras de Datos Avanzadas

Tarea Tries – José Luis Gutiérrez Espinosa (179888)

Para este trabajo se implementó con estructuras ligadas la estructura de nombre “Trie” (“Prefix tree” en inglés). Para probar su rendimiento y eficiencia, se realizó la siguiente prueba: se utilizó un banco de 100,000 palabras (obtenido del gist de github sugerido por el profesor <https://gist.github.com/h3xx/1976236>). Este banco de palabras se insertó en un arreglo en el orden en que aparecen y luego fue ordenado con el método “Merge Sort” para tener un valor al cual comparar el rendimiento de la Trie. Después, se insertaron las mismas palabras a una Trie con un arreglo de llaves generado por todos los caracteres distintos que se encontraban dentro del archivo. Se obtuvieron los siguientes resultados:

Tiempo para insertar en arreglo (para después ordenar con Merge Sort) 🡪 0.1893017 s

Tiempo para ordenar el arreglo con Merge Sort 🡪 0.1087346 s

Tiempo total para insertar en el arreglo y ordenar con Merge Sort 🡪 0.2980363 s

Tiempo para insertar de forma ordenada las palabras en la Trie 🡪 0.4601404 s

Tiempo para insertar las palabras ordenadas en un arreglo🡪0.1152948 s

Tiempo total para insertar palabras ordenadas en el arreglo usando la Trie🡪 0.5754352 s

Podemos observar que el tiempo requerido para insertar de forma ordenada en una Trie es mayor al tiempo que lleva insertar en un arreglo y luego ordenarlo por Merge Sort. Sin embargo, en esta prueba se midió el tiempo de un solo ordenamiento por Merge Sort mientras que se midió el tiempo de todas las inserciones en la Trie, por lo que la eficiencia de dicha estructura se podría apreciar en un escenario más regular, en donde cada que se insertara un dato se tuviera que ordenar el arreglo, en cuyo caso, la Trie tardaría significativamente menos.