

Algoritmos y Estructuras de Datos

Proyecto Segundo Parcial – 2018

Implementar Ordenamiento sobre lista enlazada.

A partir de un archivo de texto, genere una lista enlazada donde los nodos tengan cada palabra del archivo. El archivo debe tener al menos 1000 palabras.

Deberá implementar un ordenamiento de la misma utilizando los algoritmos de ordenamiento quick-sort y shell-sort.

El acceso a cada elemento de la lista debe ser posicional al estilo de un arreglo, es decir, un elemento solo puede ser accedido por su posición; ej: si l es una lista, $l \rightarrow \text{get}(4)$ devuelve el cuarto elemento en la lista. Para implementar el acceso posicional, deberá implementar acceso de $O(\lg n)$ (Orden logaritmico base 2) por medio de un árbol de posiciones que debe ser completo y cuya raíz tendrá como valor la mayor potencia de dos menor que el número de palabras a ordenar.

Una vez ordenada la lista deberá generarse un archivo con las palabras ordenadas de mayor a menor.

Para cada una de las opciones, deberá llevar un contador que acumule la cantidad de comparaciones realizadas por el algoritmo de ordenamiento, de forma tal que se pueda comparar el rendimientos en todos los casos.

Opción de Promoción

Para alcanzar la opción de promoción, además de lo solicitado anteriormente, deberá utilizar el mismo archivo de entrada, sin ordenar, para construir un árbol AVL. En este caso, si la palabra está repetida, debe formar parte del árbol solo una vez junto con el número de repeticiones que tiene cada palabra en el archivo original.

Como resultado deberá generar un archivo con un recorrido en pos-orden del AVL, donde figure la palabra y su número de repeticiones, la altura del arbol en que se encuentra cada palabra y contar el total de palabras incluyendo sus repeticiones. Esta suma deberá coincidir con la cantidad de palabras del archivo de entrada.