

Algoritmos y Estructuras de Datos

Proyecto Segundo Parcial – 2021

Implementar Ordenamiento Heap Sort.

La tarea consiste en ordenar alfabéticamente un archivo de texto con palabras, las cuales algunas pueden estar repetidas y otras no, como sucedería en cualquier texto normal. El archivo debe tener al menos 1000 palabras.

Deberá hacerlo con el algoritmo n-ary heap-sort basado en árboles n-arios con nodos dinámicos, generalizado para un n arbitrario.

Un heap binario es un heap que tiene 2 hijos ($n=2$) por cada nodo y la propiedad de heap se respeta entre el nodo padre y sus dos hijos, de tal forma que se cumple con:

$\text{raiz}(\text{padre}) > \text{raiz}(\text{izq}(\text{padre}))$ y
 $\text{raiz}(\text{padre}) > \text{raiz}(\text{der}(\text{padre}))$

Un heap n-ario es un heap donde cada nodo tiene n hijos y la propiedad se cumple para todos los hijos. El armado de la heap se hace igual al caso binario, completando “de izquierda a derecha” los nodos a medida que se va agregando cada dato.

En este caso deberá evitar los valores repetidos en la heap, pero cada nodo debe tener un contador de repeticiones. Es decir cuando se agrega una palabra ya existente, se incrementa en uno el contador, pero el nodo respectivo donde se insertó esta palabra queda finalmente desocupado.

La estructura de nodo del árbol deberá entonces contener un espacio para la palabra en cuestion, un contador, n-links a los n-hijos y tambien un link al padre para poder realizar los swaps respectivos que permitan mantener la propiedad de heap. Además, como no existe un valor de indexado como en el caso de utilizar un array como soporte, deberá usar una cola para indicar cual es el nodo en donde deben insertarse los hijos.

La etapa de ordenamiento final de la heap es del mismo tenor que en un monticulo binario. En este caso debe usar una pila para llevar la referencia del nodo sobre el cual se debe trabajar para realizar el ordenamiento final.

Deberá tener como entrada el orden del montículo (el valor de n) y generar un archivo de salida de las palabras ordenadas, la cantidad de repeticiones que tuvo cada una y la cantidad de comparaciones que fueron necesarias para el ordenamiento final.