

En este programa se busco leer, guardar y ordenar una bitácora haciendo uso de listas doblemente ligadas para guardar los datos y se usó Merge Sort para ordenar los datos. Esto resulto en un programa eficiente.

El uso listas doblemente ligadas es perfecto cuando se busca recorrer una lista hacia adelante o hacia atrás, además la inserción o eliminación de nuevos datos al inicio o al final de la lista es constante  $O(1)$ . Sin embargo, tiene la desventaja de no contar con índices lo que dificulta o imposibilita el uso de algoritmos de ordenamiento que necesiten acceso aleatorio aparte de aumentar la complejidad de implementación de cualquier otro, debido a los 2 apuntadores que hay en cada nodo de la lista.

Estas limitaciones nos llevo a usar el algoritmo de ordenamiento Merge Sort ya que no depende de accesos directos y cuenta con una complejidad de  $O(n \log n)$  lo que lo hace ideal para listas ligadas como la de este ejercicio.

En conclusión, aunque se haya logrado hacer un programa rápido usando listas doblemente ligadas, esta estructura no es lo ideal para casos como este, no se necesitó recorrer la lista hacia atrás y el no tener índices dificulto el ordenamiento de los datos. Para casos como este un árbol de búsqueda binaria balanceado (AVL) seria lo mejor, ya que los datos se ingresan ya ordenados.