

Instituto Tecnológico y de Estudios Superiores de Monterrey

Christian Eduardo Terrón García

A00828146

Octubre 16, 2020

Tarea 2.3

Programación de estructura de datos
y algoritmos fundamentales.



**Tecnológico
de Monterrey**

Funciones del header DoubleLinkedList.h

addFirst(Info data); ->Complejidad: $O(1)$

Añade a la lista un nuevo nodo que este al principio de la misma lista.

addLast(Info data); ->Complejidad: $O(1)$

Añade a la lista un nuevo elemento que este al final de la lista doblemente encadenada.

print(int orden); -> Complejidad: $O(n)$

Imprime toda la información de la lista basado en el parámetro para que sea de Head a Tail o viceversa

sort(); -> Complejidad: $O(n^2)$

Ordena toda la lista de menor a mayor en la lista doblemente encadenada.

search(long long rango1, long long rango2); -> Complejidad: $O(n)$;

Busca por key de cada usuario mediante búsqueda secuencial para que se despliegue la información de los nodos que cumplan el requerimiento entre el rango 1 y rango 2.

escribir(); -> Complejidad: $O(n)$

Guarda la información de la lista DoblementeEncadenada a un nuevo txt "bitacoraOrdenada.txt"

deleteAll(); -> Complejidad: $O(n)$;

Elimina todos los contenidos de la lista encadenada.

Reflexión

Entender el funcionamiento de la lista Doblemente Encadenada es de gran utilidad debido a su gran flexibilidad de moverse a través de los nodos con una mayor eficiencia debido a que podemos movernos al siguiente nodo o al anterior.

Entre otra de las ventajas es que ocupa menos procesamiento para realizar nuevas funciones que sean de utilidad para las bases de datos como ordenamiento de la información como tal.

La complejidad computacional varía entre funciones, pero se mantiene $O(n)$ como funciones para imprimir o búsqueda mientras otros para añadir información o modificar es $O(n)$.

Desventajas que tiene la Lista Doblemente encadenada es que no puedes tener un lugar o punto medio para hacer por ejemplo una búsqueda binaria que nos ayude con la búsqueda de información.

En conclusión, estas listas son altamente efectivas y nos dan una gran flexibilidad en el manejo de datos que pueden almacenar, permitiéndonos crear una gran cantidad de funciones para las bases de datos que podamos ocupar y poder integrar funciones que no nos permite un array o vector por defecto.