



Campus Querétaro

Programación de estructuras de datos y algoritmos fundamentales
(Gpo 602)

Reflexión Actividad Integral estructura de datos lineales

Presenta

David René Langarica Hernández | A01708936

Profesores

Francisco Javier Navarro Barrón

16 de octubre de 2022

Las listas doblemente enlazadas son un tipo de estructura de datos en el cual cada nodo tiene dos apuntadores, uno que apunta hacia el nodo anterior y otro que apunta al posterior. Esto, agiliza bastante la tarea de, en ciertos casos, poder navegar por todos los datos sin una dirección previamente definida, si no que es posible regresar o avanzar en esa serie de datos en particular.

Para situaciones similares a la que fue abarcada en el presente trabajo, este tipo de estructuras tienen una utilidad al nivel de otro tipo de estructuras de datos. Esto, porque se considera que las bases de datos de, en este caso, de navegación de barcos tienen un número bastante grande de entradas, en lo que se da hincapié a la necesidad de manejar los datos mediante una estructura de datos.

Las listas doblemente enlazadas son comúnmente utilizadas para los tipos de navegación en los que es preciso dirigirse hacia atrás o hacia adelante, como los navegadores hacen con los botones de anterior y siguiente para páginas web. Del mismo modo, las funciones de rehacer y deshacer, el “Most recently used (MRU)” o “Least Recently Used (LRU)” caché, y el administrador de procesos de un sistema operativo, son algunos ejemplos que se implementan mediante esta estructura de datos, por lo que son sumamente necesarias e importantes para estos (Kumari, 2022).

No obstante, y reiterando, las listas doblemente enlazadas son más empleadas para datos en los que sea necesario el regresar o avanzar desde un punto determinado, y en esta situación no es necesaria esa función, pues la necesidad de la situación es más que nada filtrar sus datos para su análisis posterior.

Anand (2021) menciona las siguientes ventajas para las listas de datos doblemente enlazadas:

- Permiten iterar en ambas direcciones.
- La eliminación de los nodos es fácil en comparación con las listas enlazadas, pues en esta se tiene acceso al nodo anterior.
- Rebobinar en los datos es más fácil.
- Su tamaño puede cambiar dinámicamente.
- Es útil para la implementación de otras estructuras de datos.

Con las ventajas anteriores, es posible afirmar que usar listas doblemente enlazadas para cualquier tipo de datos puede sonar conveniente, ya que tienes un mayor acceso a distintas funcionalidades que otros tipos de estructuras de datos no tienen. Sin embargo, las listas doblemente enlazadas también tienen ciertas desventajas, las cuales son descritas nuevamente por Anand (2021):

- Este tipo de listas consumen más memoria, pues es necesario agregar un apuntador extra en comparación con las listas enlazadas.

- Las operaciones requieren de más tiempo para arrojar resultados debido al tener que manejar un puntero extra.
- No se puede acceder aleatoriamente a los datos.

Como se puede analizar, muchas de las desventajas recaen en el consumo de memoria, pues el tener que manejar dos apuntadores para cada nodo hace menos ágil el proceso de obtener resultados esperados cuando se tienen miles y miles de nodos para una sola estructura.

Por lo cual, desde mi punto de vista, las listas doblemente enlazadas no son particularmente necesarias para solventar las necesidades como las de la presente situación, pues simplemente se necesitaría de más memoria para poder cumplir con las tareas, haciendo más lento el proceso de obtención de resultados cuando realmente no son necesarias las funciones características de las listas doblemente enlazadas.

Para este tipo de situaciones, en las que la necesidad sea meramente el filtrado y ordenamiento de los datos (sin funciones más allá de las mismas), basta con el uso de vectores o de listas simples enlazadas, pues ambas estructuras son dinámicas y permiten controlar de datos sin ocupar más memoria de la necesaria.

Finalmente, entrando en materia de la solución en código del programa, se procuró que todas las funciones que intercedían en dicho programa fueran de una complejidad $O(n)$, con el objetivo de optimizar la ejecución del programa y manteniendo el consumo de recursos lo más bajo posible. Es por ello que, de igual forma, en el programa, se utilizaron vectores para el control de los datos (asignados mediante clases).

Referencias

Anand, A. (2021, 7 de agosto). Advantages Disadvantages And Uses Of a Doubly Linked List. *PrepBytes Blog*. [Traducido del inglés]. <https://www.prepbytes.com/blog/linked-list/advantages-disadvantages-and-uses-of-a-doubly-linked-list/>

Kumari, Y. (2022, 31 de agosto). Advantages, Disadvantages And Uses Of a Doubly Linked List. *Code Studio*. [Traducido del inglés]. <https://www.codingninjas.com/codestudio/library/advantages-disadvantages-and-uses-of-a-doubly-linked-list>