Ricardo Fernández - A01704813

Reflexión Listas Doblemente Ligadas

Las listas doblemente enlazadas son una estructura de datos que consta de una lista que no solo tiene apuntadores al elemento siguiente, si no que tiene también un apuntador al nodo anterior.

Esta estructura de datos, al igual que la lista ligada simple, tiene la ventaja de ser teóricamente infinita, limitada solo por el hardware, pero sufre de las mismas desventajas como la falta de indexado. Sin embargo, esta lista tiene más aplicaciones que una lista sencilla, dado que te permite regresar al nodo anterior, sin tener que volver a recorrer la lista desde el inicio, a costa de la memoria extra usada en guardar el apuntador al nodo anterior.

En cuanto a aplicaciones reales, tenemos cosas como la lista de reproducción de aplicaciones, dado que éstas te permiten avanzar y volver dentro de la lista, sin tener un tamaño predeterminado. Otros ejemplos son las páginas de antes y después de un navegador web y sistemas de navegación.

Finalmente, esta estructura de datos también se puede utilizar para implementar otras estructuras de datos como los stacks, las queues, las hash tables o los árboles binarios.

Aplicación en la Actividad

En la actividad, yo implementé la actividad haciendo uso de vectores y arreglos, en vez de listas, lo cuál creo que es más adecuado para esta actividad. La razón de esto es que, a pesar de que no necesito el indexado como tal, la cantidad de elementos con los que vamos a trabajar está definida desde el principio, por lo que el espacio extra que conlleva el uso de listas y más aún de listas doblemente ligadas, es innecesario.

Si recorrer la lista de adelante a atrás fuera parte importante del problema, la lista doblemente ligada podría haber sido más eficiente que un arreglo. En general, es importante analizar los problemas y determinar qué estructura de datos es la

necesaria para cada situación, ya que no existe estructura de datos perfecta. Ver las ventajas y desventajas de cada una y aplicarlas a la problemática resulta esencial para un buen desempeño del programa.

Referencias

Geeks for Geeks. (2022, July 26). *Advantages, Disadvantages, and uses of Doubly Linked List*. GeeksforGeeks. Recuperado October 17, 2022, de https://www.geeksforgeeks.org/advantages-disadvantages-and-uses-of-doubly-linked-list/

Prakash, P., & Ritchie, D. (2015, October 26). *Data Structure : Doubly Linked List*. Codeforwin. Recuperado October 17, 2022, de https://codeforwin.org/2015/10/doubly-linked-list-data-structure-in-c.html