



Tecnológico de Monterrey

Evidencia 3 - Actividad Integral de Estructuras de Datos Jerárquicas

Mauricio Zavala Sánchez - A00837332

27 de octubre del 2023

David Alonso Cantú Delgado

Programación de estructuras de datos y algoritmos fundamentales (Gpo 608)

La importancia y eficiencia del uso de diferentes estructuras de datos jerárquicas, como los árboles binarios de búsqueda (BST), los árboles AVL y los Heaps, son fundamentales en la informática y la ciencia de la computación. Estas estructuras de datos son esenciales para la organización y manipulación de datos en una variedad de aplicaciones.

Árbol Binario de Búsqueda (BST):

Un árbol binario de búsqueda es una estructura de datos jerárquica que tiene la propiedad de que cada nodo tiene dos hijos, uno izquierdo y otro derecho, y los nodos se organizan de manera que los valores menores que la raíz se encuentran en el subárbol izquierdo y los valores mayores en el subárbol derecho. Esta propiedad facilita la búsqueda y la inserción de datos, ya que el proceso es similar a una búsqueda binaria.

Árbol AVL:

Un árbol AVL es una variante de un BST que está balanceado. Esto significa que la diferencia de alturas entre los subárboles izquierdo y derecho de cualquier nodo (conocido como el factor de equilibrio) no puede ser mayor que 1. Esto asegura que el árbol esté equilibrado y que las operaciones de búsqueda e inserción sean más eficientes en comparación con un BST estándar.

Heap:

Un heap es una estructura de datos jerárquica que se utiliza comúnmente para implementar colas de prioridad y es especialmente eficiente en la extracción del elemento de mayor prioridad. Los heaps pueden ser de dos tipos: heaps máximos (el elemento de mayor valor está en la raíz) y heaps mínimos (el elemento de menor valor está en la raíz).

En resumen, las estructuras de datos jerárquicas como BST, AVL y Heap son fundamentales en la informática y la ciencia de la computación. La elección de la estructura adecuada depende de las necesidades específicas de la aplicación. Los BST son eficientes para la búsqueda e inserción, los AVL garantizan un equilibrio que mejora las operaciones de búsqueda, y los heaps son ideales para mantener elementos en orden de prioridad. La comprensión de estos conceptos y algoritmos es esencial para la eficiencia en el procesamiento de datos en una amplia gama de aplicaciones.