

Bellevue University

# AVL, Red-Black, and Tries

Timothy Jelinek

CS420-T301 Data Structures

7/24/2024

When working with data, it is important to know about tree structures. Tree structures are used to store data and they store this data in a hierarchy. Properties of tree structures include nodes and edges. You can modify these trees with functions that are used to create, insert, search, and traversal. Professionals use tree structures in real world applications such as file systems. Three popular ways to use tree structures are AVL trees, red-black trees, and tries.

A popular tree structure is AVL trees. AVL trees are height balanced binary search trees. These trees being balanced is great, however, inserting and deleting elements can cause imbalance. To rebalance the trees, you can perform rotations, which there are four different rotations that are possible to be performed on them. AVL trees are unbalanced when the nodes don't have a value between -1 and 1. As long as the trees are balanced, they will work wonderfully and be useful in many different circumstances.

The second popular tree structure being talked about in this paper is red-black trees. These trees are self-balancing unlike AVL trees. Nodes that are inserted are red nodes, and they need to be recolored and rotated to keep the tree balanced. Red nodes are not allowed to be connected to other red nodes. When a black node is deleted, the result will either be double black nodes or neither red nor black. This causes an imbalance that needs to be fixed. These trees are easier to manage than AVL trees due to their self-balancing qualities.

The final subject of this paper is tries. Everyday applications use spellcheck through word retrieval and storage. This happens through nodes storing the entire

alphabet that it then uses to retrieve strings. Every trie has an empty root node. Root nodes have the same amount of links as letters in the alphabet they are coded in for spellchecking. It is very hard to delete nodes or strings in tries, so it is best to make them be futureproof. The deletion protocol involves setting every matching node to that string as null. Tries are very useful, and many people use apps for spell checking and grammar checking, some even pay for subscriptions for this, however, the maintenance of tries is complex.

This paper has gone over AVL trees, red-black trees, and tries. It is important to know about these data structures and how to use them effectively in applications such as file systems. There are operations you can perform on data sets stored in tree structures to manage and edit the data. Nodes and edges are used to identify tree structures and analyze them. It is very useful to store data in a hierarchy, in which tree structures do very well. Tree structures are great ways to store data, but it is also important to learn about other data structures. These tree structures are very useful and are great options when accomplishing tasks and creating applications. Spell checkers and grammarcheckers are used everywhere. Schools recommend students to use Grammarly when checking their grammar and spelling.

Sources:

*AVL Tree (Data Structures) - javatpoint.* (n.d.). [www.javatpoint.com](http://www.javatpoint.com).

<https://www.javatpoint.com/avl-tree>

*Red-Black tree.* (n.d.). <https://www.programiz.com/dsa/red-black-tree>

Joshi, V. (2018, June 20). Trying to Understand Tries - basecs - Medium. *Medium*.

<https://medium.com/basecs/trying-to-understand-tries-3ec6bede0014>

GeeksforGeeks. (2024, July 12). *Introduction to tree data Structure*. GeeksforGeeks.

<https://www.geeksforgeeks.org/introduction-to-tree-data-structure/>