Daee kang

Ass 5

Sec 1004

Avl tree summary:

An AVL tree is binary search tree that is simply balanced: binary search tree meaning that it will have two children with no duplicate nodes and the left will always be less than the parent and vise-versa. The AVL tree will include rotate functions that will keep always balanced.

Advantages:

AVL tree is mainly beneficial due to the fact that they are rigidly balanced. The balance will always be based on height and will aim to have the smallest height. With a smaller height, we can keep a true big O for searching at log(n). This makes searching more effective than searching through a binary tree where the big O can be O(n).

However, due to the fact that it always has to be balanced, there is a lot more overhead to keep it balanced, which requires after every insert, a rotation operation be used if needed. However, it's a sacrifice to keep search times low.

O notation for AVL ops:

Insert: O(logn)

Delete:O(logn)

Search:O(logn)

Rotate: O(logn)