***AVL Trees***

AVL trees are binary search trees also commonly called as “Height balanced tree” meaning we can control the depth of the tree.

The left subtree and right subtree are height balanced themselves and the dept of the left side and the depth of the right side can differ at most by 1

For eg :

**Case 1**

Violation at the right child of right subtreeA picture containing object, clock

Description automatically generated

To remove the violation we rotate the grandparent and we do a left rotation and we end up with

**Case 2**

Violation at the left child of left subtree

A screenshot of a cell phone

Description automatically generated If we have 2 children on a left of the left subtree We will do a right rotation of the node

**Case 3**A picture containing clock

Description automatically generated

Violation at the left child of right subtree

We do a left right rotation of the grandparent of the problem node

Then we do a left rotation of the parent of the problem  node

Then we do the right rotation of the grandparent of the problem node.

Then we finally do left rotation of the grandparent of the problem node.

**Case 4:**

A screenshot of a cell phone

Description automatically generated

***Splay Trees***

Type of binary search tree but they are not strictly balanced (nearly balanced) but that's why they are relatively very faster.

1. It provides fast access to elements that we have recently used.
2. Splay tree operations run in O(log n) time on average
3. In addition to search it also splays (moves the node to the root). If the node is found
4. Then it is splayed and it becomes the new node. Else the last node before reaching the null is splayed and made the new root
5. We follow the same pattern with insert
6. For delete we follow the same procedure and

* If we find do not find the node we wanted to delete we splay the node that ended our search
* If we find the node that we wanted to delete we remove the key and then we splay the parent up to the root.

What splaying does is it allows the tree to balance itself.

PICTURES TAKEN FROM GEEKSFORGEEKS