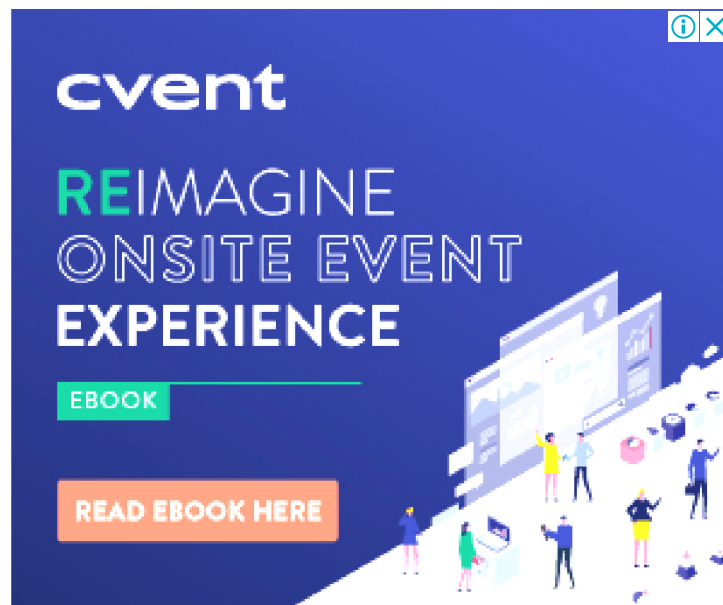


AVL Trees

Tree is one of the most important data structure that is used for efficiently performing operations like insertion, deletion and searching of values. However, while working with a large volume of data, construction of a well-balanced tree for sorting all data is not feasible. Thus only useful data is stored as a tree, and the actual volume of data being used continually changes through the insertion of new data and deletion of existing data. You will find in some cases where the NULL link to a binary tree to special links is called as threads and hence it is possible to perform traversals, insertions, deletions without using either stack or recursion. In this chapter, you will learn about the Height balance tree which is also known as the AVL tree.



What is The AVL Tree?

AVL tree is a binary search tree in which the difference of heights of left and right subtrees of any node is less than or equal to one. The technique of balancing the height of binary trees was developed by Adelson, Velskii, and Landi and hence given the short form as AVL tree or Balanced Binary Tree.

An AVL tree can be defined as follows:

Let T be a non-empty binary tree with T_L and T_R as its left and right subtrees. The tree is height balanced if:

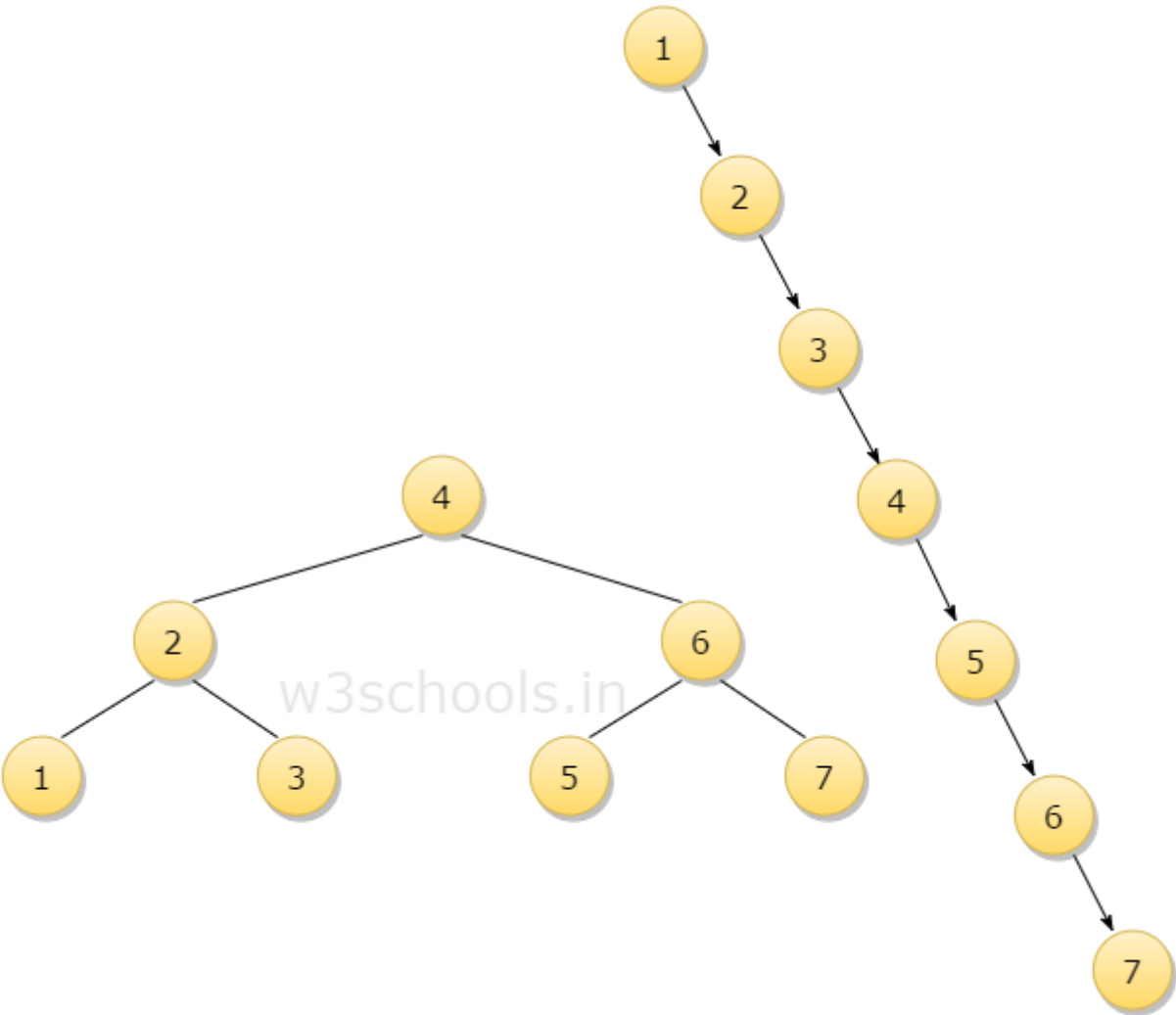
- T_L and T_R are height balanced
- $h_L - h_R \leq 1$, where h_L - h_R are the heights of T_L and T_R

The Balance factor of a node in a binary tree can have value 1, -1, 0, depending on whether the height of its left subtree is greater, less than or equal to the height of the right subtree.

Advantages of AVL tree

Since AVL trees are height balance trees, operations like insertion and deletion have low time complexity. Let us consider an example:

If you have the following tree having keys 1, 2, 3, 4, 5, 6, 7 and then the binary tree will be like the second figure:



To insert a node with a key Q in the binary tree, the algorithm requires seven comparisons, but if you insert the same key in AVL tree, from the above 1st figure, you can see that the algorithm will require three comparisons.

| | |
|---------------------|----------------------------|
| Tree Removal Sydney | Affordable Tree Removal |
| Cheap Tree Service | Flowering Ornamental Trees |
| Tree Removal Prices | Free Tree Removal |
| Tree Removal Cost | Shade Trees |
| Pictures Of Trees | Upside-Down Christmas Tree |

Representation of AVL Trees

```
Struct AVLNode
{
    int data;
    struct AVLNode *left, *right;
    int balfactor;
};
```

Algorithm for different Operations on AVL

For Insertion:

- Step 1: First, insert a new element into the tree using BST's (Binary Search Tree) insertion logic.
- Step 2: After inserting the elements you have to check the Balance Factor of each node.
- Step 3: When the Balance Factor of every node will be found like 0 or 1 or -1 then the algorithm will proceed for the next operation.
- Step 4: When the balance factor of any node comes other than the above three values then the tree is said to be imbalanced. Then perform the suitable Rotation to make it balanced and then the algorithm will proceed for the next operation.

For Deletion:

- Step 1: Firstly, find that node where k is stored

Step 2: Secondly delete those contents of the node (Suppose the node is x)

Step 3: Claim: Deleting a node in an AVL tree can be reduced by deleting a leaf. There are three possible cases:

- When x has no children then, delete x
- When x has one child, let x' becomes the child of x.
- Notice: x' cannot have a child, since subtrees of T can differ in height by at most one :
 - then replace the contents of x with the contents of x'
 - then delete x' (a leaf)
- Step 4: When x has two children,
 - then find x's successor z (which has no left child)
 - then replace x's contents with z's contents, and
 - delete z

In all of the three cases, you will end up removing a leaf.

| | |
|-------------------------|-----------------------------|
| Big Data Courses | Cloud Computing Training |
| Create A Website | HTML Programming Courses |
| Big Data Certifications | Earn Money Online |
| Online Master's Degrees | Java For Beginners |
| Java Programmer Jobs | Java Certification Training |

[Forests and Orchards >](#)

SPONSORED SEARCHES

data analysis r

Q

analyze data

Q

analyzing data

Q

usmle step 1

Q

