

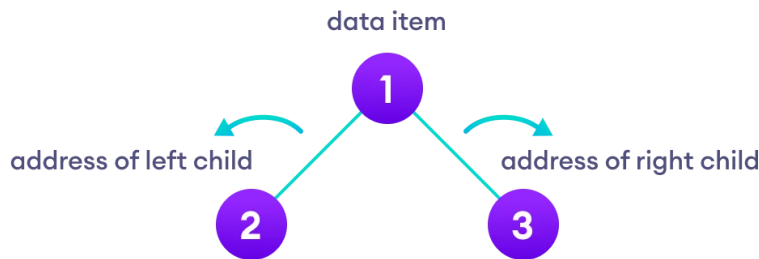
Binary Tree Documentation

A binary tree is a tree data structure whose all nodes have either zero, one, or at most two children nodes. These two children are generally referred to as left and right children respectively.

The top-most node is known as the root node, while the nodes with no children are known as leaf nodes.

Each node in a binary tree has these three elements –

1. Data item that is the data stored in the node
2. Address of the left child
3. Address of the right child



Terminologies in Binary Tree in Data Structure

1. Nodes – Nodes are the building blocks of any data structure. They majorly contain some data and link to the next/previous nodes. In the case of binary trees, they contain the address of the left and the right child respectively.

2. Root – The topmost node in a tree is known as the root node. A tree can have at most one root node.

3. Parent Node – A node (except the root) that has a succeeding node is known as a parent node.

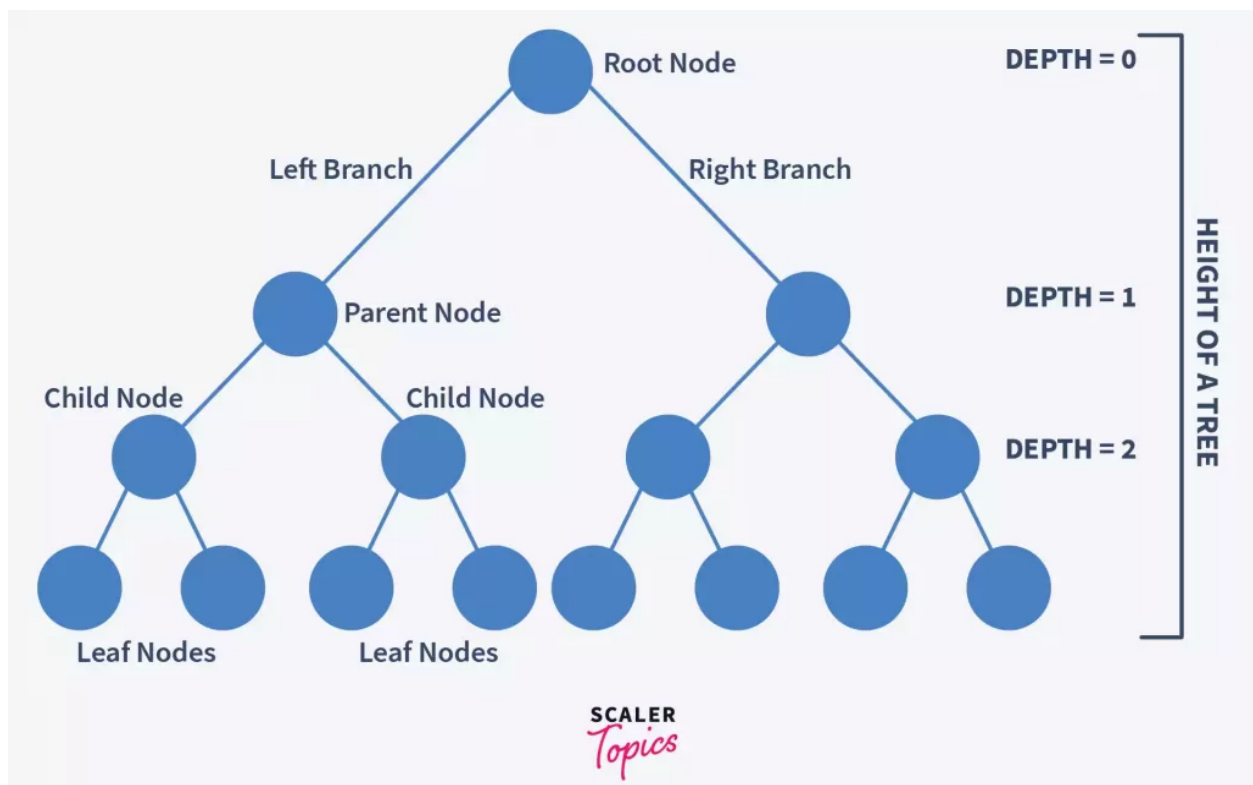
4. Child Node – A node that has a preceding node is known as a child node. A node can be both parent and child depending on the node that is in context.

5. Leaf Node – A node with no children.

6. Internal Node – A node that has at least one child node is known as an internal node.

7. Depth of a Binary Tree – The number of edges from a node in the tree to the root node.

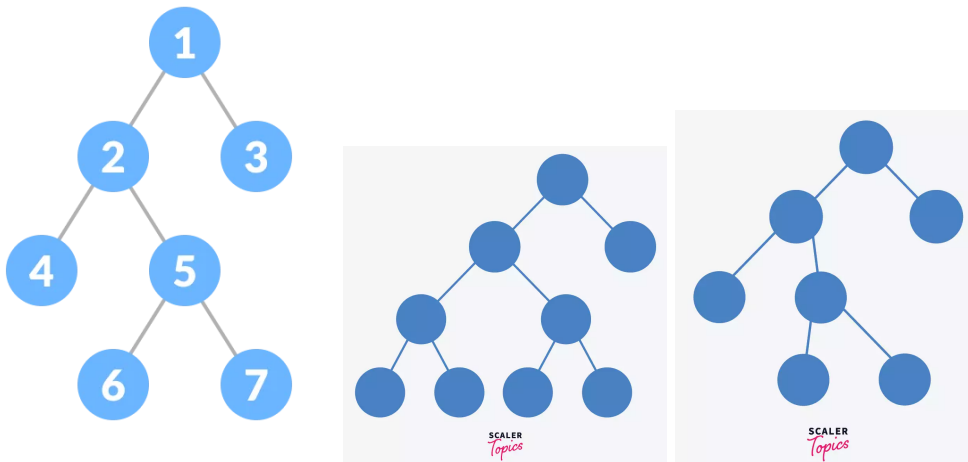
8. Height of a Binary Tree – The number of edges from the deepest node in the tree to the root node.



Types of Binary Tree

1. Full Binary Tree

A full Binary tree is a special type of binary tree in which every parent node/internal node has either two or no children.

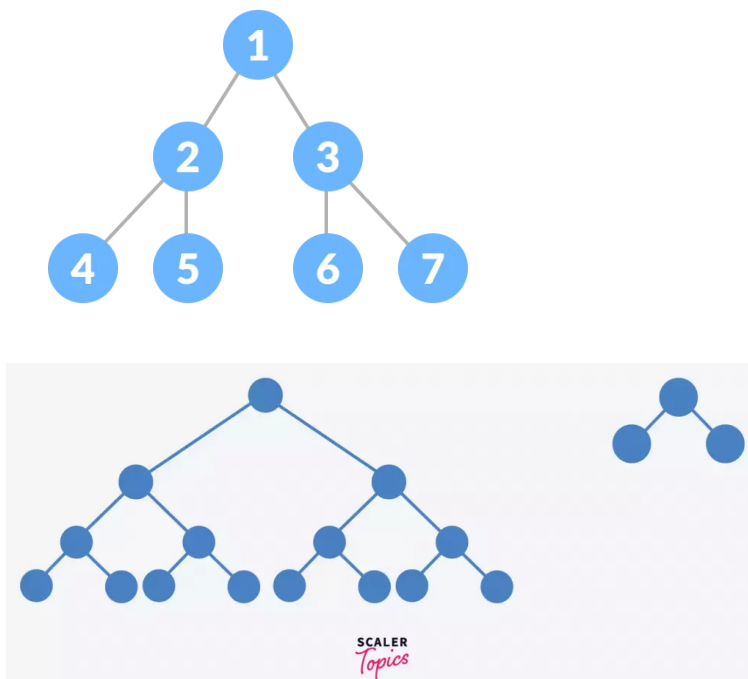


In a full binary tree, if there are n number of total nodes

- The number of leaf nodes is given by $(n+1)/2$

2. Perfect Binary Tree

A perfect binary tree is a type of binary tree in which every internal node has **exactly two child nodes** and all the leaf nodes are at the same level.

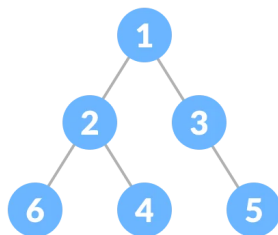
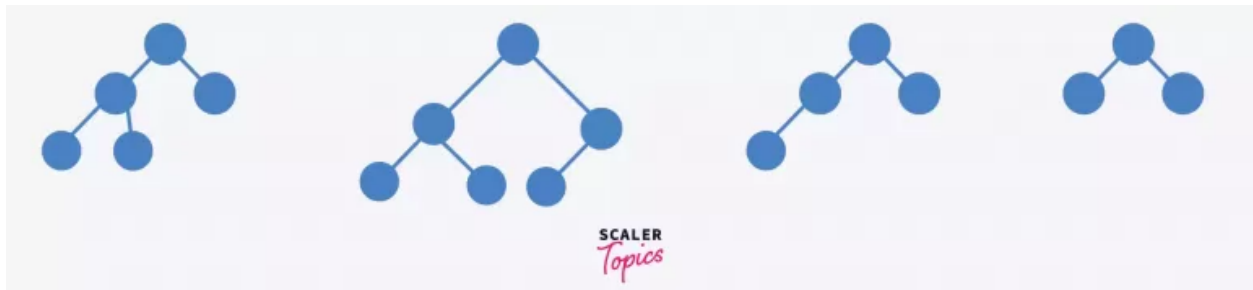


Consider a perfect binary tree with height h , the total number of nodes in this case is given by $2^h - 1$.

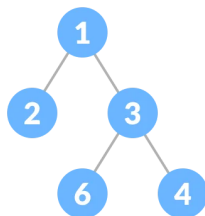
3. Complete Binary Tree

A complete binary tree is just like a full binary tree, but with two major differences

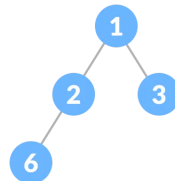
1. Every level must be completely filled
2. All the **leaf elements must lean towards the left.**
3. The **last leaf element might not have a right sibling** i.e. a complete binary tree doesn't have to be a full binary tree.



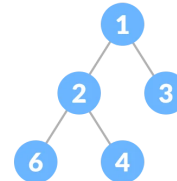
✗ Full Binary Tree
✗ Complete Binary Tree



✓ Full Binary Tree
✗ Complete Binary Tree



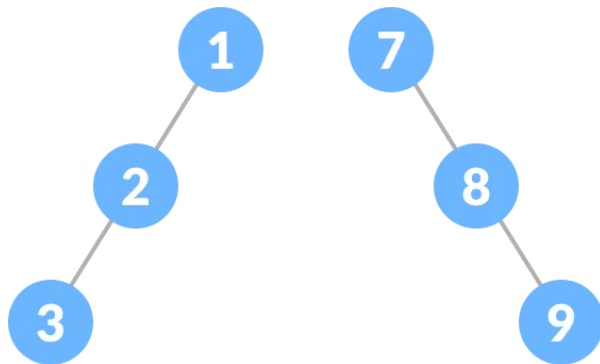
✗ Full Binary Tree
✓ Complete Binary Tree



✓ Full Binary Tree
✓ Complete Binary Tree

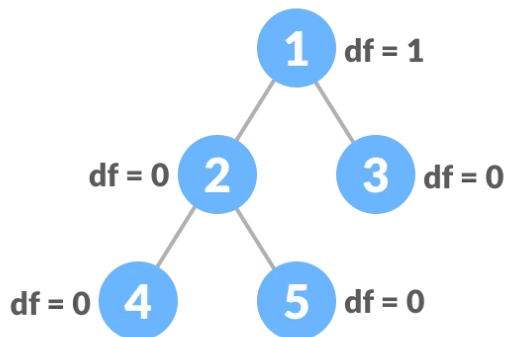
4. Skewed Binary Tree

A skewed binary tree is a pathological/degenerate tree in which the tree is either dominated by the left nodes or the right nodes. Thus, there are two types of skewed binary tree: left-skewed binary tree and right-skewed binary tree.



5. Balanced Binary Tree

It is a type of binary tree in which the difference between the height of the left and the right **subtree for each node** is either 0 or 1.



Simple Problems :

1. <https://leetcode.com/problems/root-equals-sum-of-children/description/>
2. <https://leetcode.com/problems/evaluate-boolean-binary-tree/description/>
3. <https://leetcode.com/problems/invert-binary-tree/description/>
4. <https://leetcode.com/problems/maximum-depth-of-binary-tree/description/>
5. <https://leetcode.com/problems/univalued-binary-tree/description/>