

# Data Structures : Trees

Trainer : Nisha Dingare

Email : nisha.dingare@sunbeaminfo.com



# Trees :

- It is an advanced / non-linear data structure, which is a collection/list of finite number of logically related similar type of data elements in which, there is a first specially designated element referred as root element, and remaining all elements are connected to it in a hierarchical manner follows parent-child relationship.
- In tree data structure, element is also referred as a node.
  - - Node : Item storing information and branches to other nodes.
  - - root node => first specially designated element in a tree.
  - - parent node/father
  - - child node/son
  - - grand parent/grand father
  - - grand child/grand son
  - - Siblings - Children of the same parents
  - - Ancestors :
    - all the nodes which are in the path from root node to that node.
    - root node is an ancestor of all the nodes.
  - Descendants :
    - All the nodes which can be accessible from it.
    - All the nodes are descendent of root node.



# Tree Terminologies :

- **Degree of a node** : no. of child nodes
- **Degree of a tree** : max degree of any node in a given tree
- **Leaf node** : node which is not having any number of child node/s OR node having degree 0.
- **Non-leaf node** : node which is having any number of child node/s OR node having non-zero degree
- **NULL Tree (Empty Tree)** : Tree with no node. It has only the data part.
- **Level of a node** :
  - Indicates the position of the node in the hierarchy.
  - Level of any node is level of its parent +1.
  - Level of root is 1.
  - Level of a tree -> max level of any node in a given tree.
- **Depth of a node** :
  - Number of nodes from the root to the node.
  - Depth of root is 0.
  - Level = Depth + 1.
- **Height of a node** :
  - Number of nodes from the node to its deepest leaf.
  - Height of node =  $\max(\text{height of its any child}) + 1$ .
  - Height of empty/null tree is -1.
  - Height of a tree: Height of root of the tree.
- **Traversal** : Visiting each node of tree exactly once.



# Types of Trees :

- As tree is a dynamic data structure any node can have any no. of child nodes. tree can grow up to any level.
- On such trees, operations like addition, deletion, searching etc.. become inefficient, so restrictions can be applied on it, and hence there are different types of tree.
- **Binary tree** – it is a type of tree in which each node can have max 2 no. of child nodes, i.e. each node can have either 0, 1 or 2 no. of child nodes. OR we can say it is a type of tree where each node can have degree either 0 or 1 or 2.
  - It is set of finite no. of elements having 3 subsets
    1. root node
    2. left subtree (may be empty)
    3. right subtree (may be empty)
- **Binary search tree (bst)** – it is a binary tree in which nodes are arranged as per their values. In this, left child is always smaller than its parent and right child is always greater or equal to its parent.



# Binary Search Tree Traversal :

- In-order :
  - Left
  - Parent
  - Right
- Pre-Order :
  - Parent
  - Left
  - Right
- Post-Order :
  - Left
  - Right
  - Parent



---

# Thank You !

