Tree Traversal

Tree Traversal

- Displaying (or) visiting order of nodes in a binary tree is called as Binary Tree Traversal.
- There are three types of binary tree traversals.

1. In - Order Traversal
2. Pre - Order Traversal
3. Post - Order Traversal

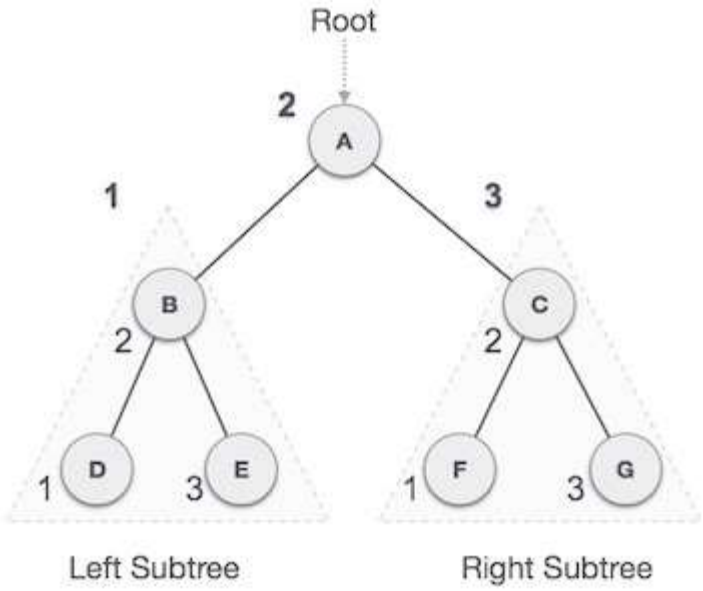
In-order Traversal

- In this traversal method, the left subtree is visited first, then the root and later the right sub-tree.

Example:

- We start from **A**, and following in-order traversal, we move to its left subtree **B**.
- **B** is also traversed in-order.
- The process goes on until all the nodes are visited.
- The output of inorder traversal of this tree will be

D → B → E → A → F → C → G



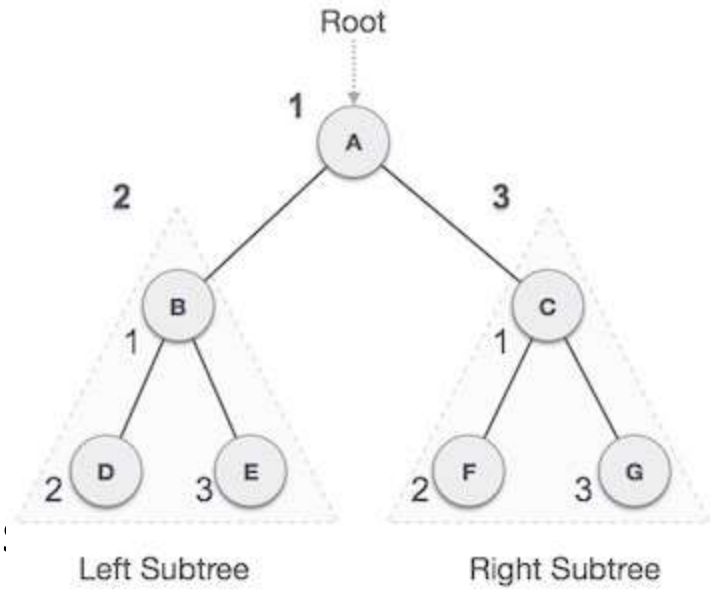
Inorder traversal

Algorithm

- Until all nodes are traversed −
- **Step 1** − Recursively traverse left subtree.
- **Step 2** − Visit root node.
- **Step 3** − Recursively traverse right subtree.

Pre-order Traversal

- In this traversal method, the root node is visited first, then the left subtree and finally the right subtree.



Algorithm:

- Until all nodes are traversed –
- **Step 1** – Visit root node.
- **Step 2** – Recursively traverse left subtree.
- **Step 3** – Recursively traverse right

We start from **A**, and following pre-order traversal, we first visit **A** itself and then move to its left subtree **B**.

OUTPUT : ***A → B → D → E → C → F → G***

Post-order Traversal

- In this traversal method, the root node is visited last, hence the name. First we traverse the left subtree, then the right subtree and finally the root node.

Algorithm

Until all nodes are traversed −

Step 1 − Recursively traverse left subtree.

Step 2 − Recursively traverse right subtree.

Step 3 − Visit root node.

We start from **A**, and following Post-order traver
subtree **B**.

OUTPUT:

D → E → B → F → G → C → A