Binary Tree

- A Binary Tree is a finite number of elements that is either empty or is partitioned into 3 sub sets.
- The first sub set contains a single element called the Root of the Tree.
- The other two sub sets are themselves Binary trees, called Left and Right Sub-Trees of the Original tree ***OR***
- A Tree is said to be Binary if each node in the Tree is associated with 0, 1 or at most 2 children nodes.
- Two Child Nodes are called the Left and Right Children nodes of the Tree.

Binary Tree Traversal

There are three Traversing techniques for Binary Trees.

1. **In-Order Traversal:**
    1.1. Visit Left-Sub Tree
    1.2. Process Parent Node
    1.3. Visit Right-Sub Tree
2. **Pre-Order Traversal:**
    2.1. Process Parent Node
    2.2. Visit Left-Sub Tree
    2.3. Visit Right-Sub Tree
3. **Post-Order Traversal:**
    3.1. Visit Left-Sub Tree
    3.2. Visit Right-Sub Tree
    3.3. Process Parent Node

Construction of a Binary Tree from a List of Numbers

A Binary Tree can be constructed from a list of numbers. The Algorithm for it is given below.

1. Whatever is the first number of the list becomes the Root of the Binary Tree.

2. if (Value-of-Child-Node-Element >= Value-of-Parent-Node-Element)
   then
   Make new number the Left-Child-Node
   else
   Make new number the Right-Child-Node

3. Verify Tree using In-Order-Traversal