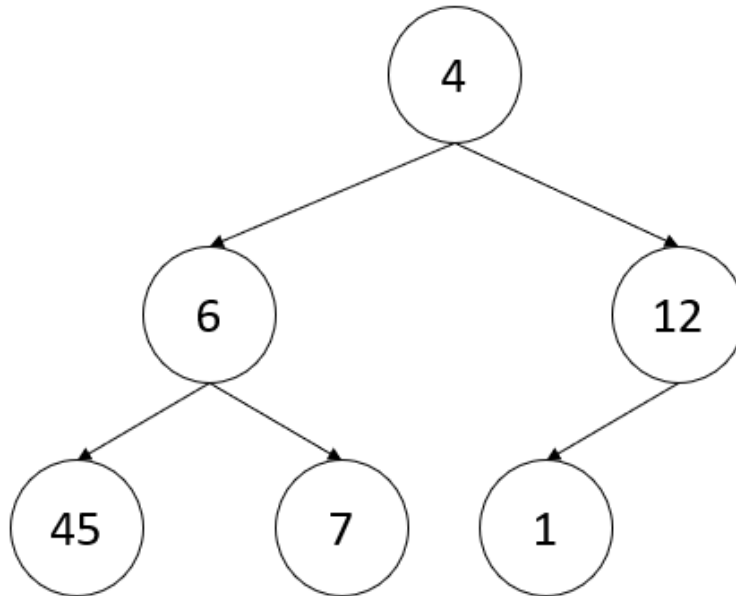


**Objectives:**

- Tree, Binary Tree, Binary Search Tree, Traversal Algorithms

**Question 1:** Write a program to build binary tree option 1 (Complete BT) which is given in Figure 1 using struct. Then write function for tree traversal algorithms which are pre-order, in-order, and post-order. In main, you fill the tree with order inputs which are “4, 6, 12, 45, 7, 1” via “**insert**” function written by you. And with inputs order, your tree must have the same structure which is given Figure 1. Expected output is given in below.



Binary Tree Option 1 (Complete BT)

**Figure 1.** Binary Tree Option 1 Structure

**Example output:**

```

preorder traversal option 1:  4 6 45 7 12 1
Inorder traversal option 1:   45 6 7 4 1 12
Postorder traversal option 1: 45 7 6 1 12 4
  
```

**Example struct:**

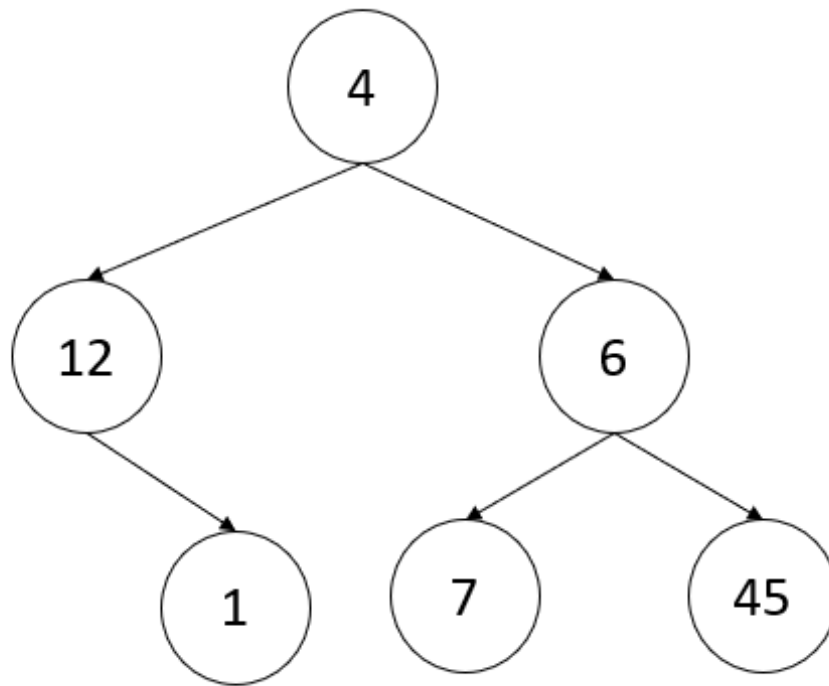
```

struct node {

    int data;
    struct node *left; //Reference to left child
    struct node *right; //Reference to right child

};
  
```

**Question 2:** Write a program to build binary tree option 2 which is given in Figure 2 using struct. Then write function for tree traversal algorithms which are pre-order, in-order, and post-order. In main, you fill the tree with order inputs which are “4, 6, 12, 45, 7, 1” via “**insert**” function written by you. And with inputs order, your tree must have the same structure which is given Figure 2. Expected output is given in below.



Binary Tree Option 2

Figure 2. Binary Tree Option 2 Structure

Example output:

```
preorder traversal option 2: 4 12 1 6 7 45
Inorder traversal option 2: 12 1 4 7 6 45
Postorder traversal option 2: 1 12 7 45 6 4
```

Example struct:

```
struct node {
    int data;
    struct node *left; //Reference to left child
    struct node *right; //Reference to right child
};
```

**Question 3:** Write a program to build binary search tree (BST) from using results of any of the traversal methods which are pre-order, in-order, post-order, and breadth first search. Example inputs and for these inputs expected output for binary search tree structure given in the below. Hint: You can learn the accuracy of the obtained tree by comparing the result you will get by using the traversal method in obtained tree and the input value.

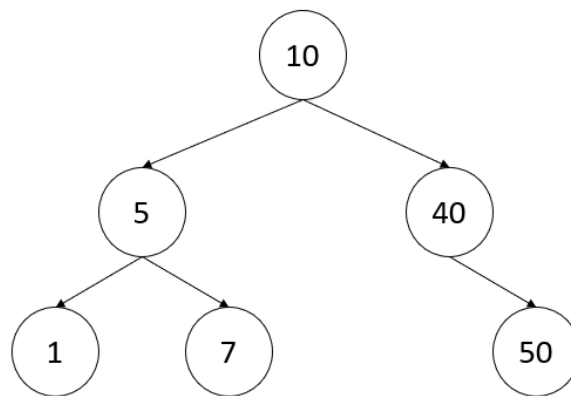
**Example input:**

Input example 1: Pre-order sequence = 10, 5, 1, 7, 40, 50

Input example 2: Breadth first search sequence = 7, 4, 12, 3, 6, 8, 1, 5, 10

**Example output:**

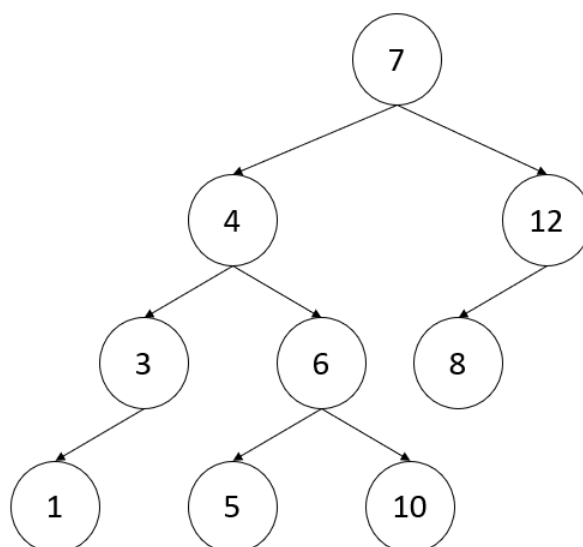
Output for input example 1:



Binary Search Tree

**Figure 3.** Binary Search Tree for input example 1

Output for input example 2:



Binary Search Tree

**Figure 4.** Binary Search Tree for input example 2