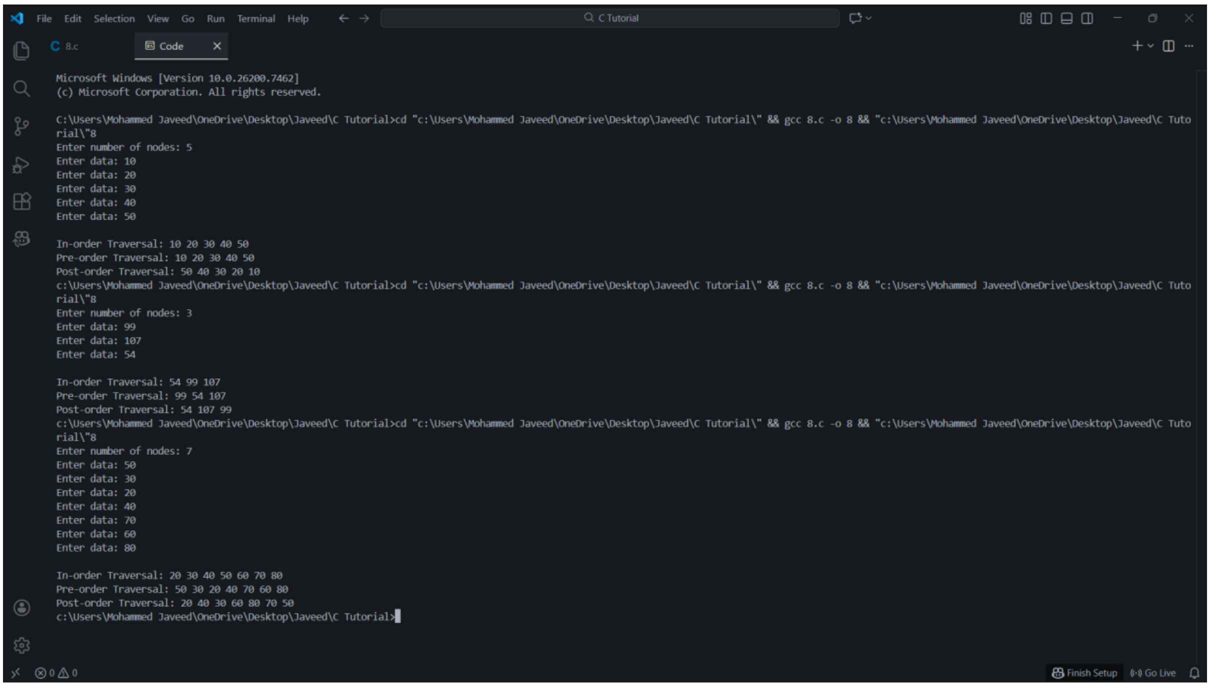
8. Write a program a) ToconstructabinarySearchtree. b) To traverse the tree using all the methods i.e., in-order, preorder and post order c) To display the elements in the tree.

```
#include <stdio.h>

#include <stdlib.h>


/* Definition of BST Node */

struct node {

   int data;

   struct node *left;

   struct node *right;

};

struct node* createNode(int data) {

   struct node *newNode = (struct node*)malloc(sizeof(struct node));

   newNode->data = data;

   newNode->left = NULL;

   newNode->right = NULL;

   return newNode;

}

struct node* insert(struct node *root, int data) {

   if (root == NULL)

      return createNode(data);


   if (data < root->data)

      root->left = insert(root->left, data);

   else if (data > root->data)

      root->right = insert(root->right, data);


   return root;

}


/* In-order Traversal */

void inorder(struct node *root) {

   if (root != NULL) {
        inorder(root->left);
        printf("%d ", root->data);
        inorder(root->right);
    }
}

/* Pre-order Traversal */
void preorder(struct node *root) {
    if (root != NULL) {
        printf("%d ", root->data);
        preorder(root->left);
        preorder(root->right);
    }
}

/* Post-order Traversal */
void postorder(struct node *root) {
    if (root != NULL) {
        postorder(root->left);
        postorder(root->right);
        printf("%d ", root->data);
    }
}

/* Main function */
int main() {
    struct node *root = NULL;
    int n, data, i;

    printf("Enter number of nodes: ");
    scanf("%d", &n);
    for (i = 0; i < n; i++) {
        printf("Enter data: ");
        scanf("%d", &data);
        root = insert(root, data);
    }

    printf("\nIn-order Traversal: ");
    inorder(root);

    printf("\nPre-order Traversal: ");
    preorder(root);

    printf("\nPost-order Traversal: ");
    postorder(root);

    return 0;
}
```

OUTPUT:



```
Microsoft Windows [Version 10.0.26200.7462]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tutorial>cd "c:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tutorial\" && gcc 8.c -o 8 && "c:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tuto
rial\"8
Enter number of nodes: 5
Enter data: 10
Enter data: 20
Enter data: 30
Enter data: 40
Enter data: 50

In-order Traversal: 10 20 30 40 50
Pre-order Traversal: 10 20 30 40 50
Post-order Traversal: 50 40 30 20 10
c:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tutorial>cd "c:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tutorial\" && gcc 8.c -o 8 && "c:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tuto
rial\"8
Enter number of nodes: 3
Enter data: 99
Enter data: 107
Enter data: 54

In-order Traversal: 54 99 107
Pre-order Traversal: 99 54 107
Post-order Traversal: 54 107 99
c:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tutorial>cd "c:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tutorial\" && gcc 8.c -o 8 && "c:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tuto
rial\"8
Enter number of nodes: 7
Enter data: 50
Enter data: 30
Enter data: 20
Enter data: 40
Enter data: 70
Enter data: 60
Enter data: 80

In-order Traversal: 20 30 40 50 60 70 80
Pre-order Traversal: 50 30 20 40 70 60 80
Post-order Traversal: 20 40 30 60 80 70 50
c:\Users\Mohammed Javeed\OneDrive\Desktop\Javeed\C Tutorial>
```