Lab PROGRAM - 10

```
#include <stdio.h>
#include <stdlib.h>
struct node
{
    int data;
    struct node *left;
    struct node *right;
};root1;

struct node* create()
{
    struct node *temp;
    printf("Enter Root element: \n");
    temp = (struct node*)malloc(sizeof(struct node));
    scanf("%d", &temp->data);
    temp->left = temp->right = NULL;
    return temp;
}

void insert(struct node *root, struct node *temp)
{
    if(temp->data < root->data)
    {
        if(root->left != NULL)
            insert(root->left, temp);
        else
            root->left = temp;
    }
    if(temp->data > root->data)
    {
        if(root->right != NULL)
            insert(root->right, temp);
        else
```

```
            root -> right = temp;
        }
    }

void printPostorder (struct node * node)
{
    if (node == NULL)
        return;
    printPostorder (node -> left);
    printPostorder (node -> right);
    printf ("%d\t", node -> data);
}
void printInorder (struct node *node)
{
    if (node == NULL)
        return;
    printInorder (node -> left);
    printf ("%d\t", node -> data);
    printInorder (node -> right);
}

void printPreorder (struct node *node)
{
    if (node == NULL)
        return;
    printf ("%d\t", node -> data);
    printPreorder (node -> left);
    printPreorder (node -> right);
}

int main ()
{
    int choice;
    struct node *temp;
    do
    {
```

```
printf("\n--- MENU ---");
printf("1. CREATE \n");
printf("2. Insert \n");
printf("3. Preorder Traversal \n");
printf("4. Inorder Traversal \n");
printf("5. Postorder Traversal \n");
printf("6. EXIT \n");
printf("Enter your choice correctly: \n");
scanf("%d", &choice);
switch(choice)
{
    case 1: root1 = Create();
            break;
    case 2: printf("Enter value to Insert:");
            temp = (struct node *) malloc(sizeof
                                (struct node));
            scanf("%d", &temp->data);
            insert(root1, temp);
            break;
    case 3: printPreorder(root1);
            break;
    case 4: printInorder(root1); break;
    case 5: printPostorder(root1); break;
    case 6: printf("EXITING....!! \n");
            break;
    default: printf("Incorrect choice! \n");
}
while(choice != 6);
return 0;
}
```