LAB PROGRAM 8

Q) Write a program to

a) to construct a binary search tree

b) to traverse tree using in,pre,post order

c)display elements in a tree

CODE:

```
#include<stdio.h>
#include<stdlib.h>
struct node
{
    int data;
    struct node *left,*right;
};
struct node* create(int value)
{
    struct node* newnode=(struct node*)malloc(sizeof(struct node));
    newnode->data=value;
    newnode->left=newnode->right=NULL;
    return newnode;
};
struct node * insert(struct node *root,int value)
{

    if(root==NULL)
        return create(value);
    if(value<root->data)
        root->left=insert(root->left,value);
    else if(value>root->data)
        root->right=insert(root->right,value);
    return root;

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

```c
void preorder(struct node *root)
{
    if(root!=NULL)
    {
        printf("%d\t",root->data);
        preorder(root->left);
        preorder(root->right);
    }
}
void postorder(struct node *root)
{
    if(root!=NULL)
    {

        postorder(root->left);
        postorder(root->right);
        printf("%d\t",root->data);
    }
}

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

int main()
{
    struct node*root = NULL;
    int n,value,i;
    printf("\nEnter the number of elements:");
    scanf("%d",&n);
    printf("\nEnter the elements:");
    for(i=0;i<n;i++)
    {
        scanf("%d",&value);
        root=insert(root,value);
    }
    printf("\nInorder traversal:");
    inorder(root);
    printf("\nPreorder traversal:");
    preorder(root);
    printf("\nPostorder traversal");
    postorder(root);
```

```
    return 0;
}
```

OUTPUT

```
Enter the number of elements:5

Enter the elements:30
40
20
10
45

Inorder traversal:10    20      30      40      45
Preorder traversal:30   20      10      40      45
Postorder traversal10   20      45      40      30
Process returned 0 (0x0)   execution time : 12.268 s
Press any key to continue.
```