#include<stdio.h>

#include<stdlib.h>

struct node

{

struct node \*left;

int data;

struct node \*right;

}\*root=NULL,\*newnode;

struct node \*insert(struct node \*root, int ele) {

if (root == NULL) {

newnode = (struct node \*)malloc(sizeof(struct node));

newnode->data = ele;

newnode->left = NULL;

newnode->right = NULL;

return newnode;

} else if (ele > root->data) {

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

} else if (ele < root->data) {

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

}

return root;

}

void inorder(struct node \*root)

{

if(root!=NULL)

{

inorder(root->left);

printf("%d\t",root->data);

inorder(root->right);

}

}

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);

}

}

int main() {

struct node\* root = NULL;

root = insert(root, 85);

insert(root, 100);

insert(root, 22);

insert(root, 50);

insert(root, 10);

insert(root, 110);

insert(root, 90);

printf("Preorder traversal: ");

preorder(root);

printf("\nInorder traversal: ");

inorder(root);

printf("\nPostorder traversal: ");

postorder(root);

    return 0;

}