#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

struct node

{

int data;

struct node \*right, \*left;

}\*root,\*p,\*q;

struct node \*make(int y)

{

struct node \*newnode;

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

newnode->data=y;

newnode->right=newnode->left=NULL;

return(newnode);

}

void left(struct node \*r,int x)

{

if(r->left!=NULL)

printf("\n Invalid !");

else

r->left=make(x);

}

void right(struct node \*r,int x)

{

if(r->right!=NULL)

printf("\n Invalid !");

else

r->right=make(x);

}

void inorder(struct node \*r)

{

if(r!=NULL)

{

inorder(r->left);

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

inorder(r->right);

}

}

void preorder(struct node \*r)

{

if(r!=NULL)

{

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

preorder(r->left);

preorder(r->right);

}

}

void postorder(struct node \*r)

{

if(r!=NULL)

{

postorder(r->left);

postorder(r->right);

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

}

}

void create()

{

int no;

char choice='y';

printf("\nEnter the root:");

scanf("%d",&no);

root=make(no);

p=root;

while(choice=='y'||choice=='Y')

{

printf("\nEnter number:");

scanf("%d",&no);

if(no==-1)

break;

p=root;

q=root;

while(no!=p->data && q!=NULL)

{

p=q;

if(no<p->data)

q=p->left;

else

q=p->right;

}

if(no<p->data)

{

printf("Left branch of %d is %d\n",p->data,no);

left(p,no);

}

else

{

right(p,no);

printf("Right Branch of %d is %d\n",p->data,no);

}

printf("Continue ?: ");

scanf(" %c",&choice);

}

}

void main()

{

int no,action;

clrscr();

while(1)

{

printf("\n\n1) Create\n");

printf("2) Inorder Traversal\n");

printf("3) Preorder Traversal\n");

printf("4) Postorder Traversal\n");

printf("5) Exit\n");

printf("Enter choice : ");

scanf("%d",&action);

switch(action)

{

case 1:

create();

break;

case 2:

printf("\n\nInorder Traversal is: \n");

inorder(root);

break;

case 3:

printf("\n\nPreorder Traversal is: \n");

preorder(root);

break;

case 4:

printf("\n\nPostorder Traversal is: \n");

postorder(root);

break;

case 5:

exit(0);

break;

}

getch();

}

}