```
#include<stdio.h>
#include<conio.h>
#include<malloc.h>
struct NODE
{
	int data;
	struct NODE *left;
	struct NODE *right;
};
typedef struct NODE node;
node *root=NULL;
void create(node *tree, node *n)
{
	int ch;
	printf("\nWant to insert left(press 1) or right (press 2) of parent %d ",tree->data);
	scanf("%d",&ch);
	if(ch==2)
	{
		if(tree->right==NULL)
			tree->right=n;
		else
			create(tree->right,n);
	}
	else
	if(ch==1)
```

```
        {
                if(tree->left==NULL)
                        tree->left=n;
                else
                        create(tree->left,n);
        }

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

```
}
void postorder(node * p)
{
	if(p!=NULL)
	{
		postorder(p->left);
		postorder(p->right);
		printf("%d ",p->data);
	}
}
void main()
{
	int choice;
	char ca,cho;
	clrscr();
	do
	{
	printf("\nEnter choice: \t1.INSERT \t2.INORDER \t3.PRE ORDER \t4.POST ORDER ");
	scanf("%d",&choice);
	switch(choice)
	{
		case 1: do
			{
				node *start;
				start=(node*)malloc(sizeof(node));
```

```
                printf("\nEnter data  ");
                scanf("%d",&start->data);
                start->left=start->right=NULL;
                if(root==NULL)
                        root=start;
                else
                        create(root,start);
                printf("\nWant to enter more elements?(y/n) ");
                fflush(stdin);
                scanf("%c",&cho);
        }while(cho=='y'||cho=='Y');
        break;
case 2: if(root==NULL)
                printf("\nTREE IS EMPTY.");
        else
                inorder(root);
        break;
case 3: if(root==NULL)
                printf("\nTREE IS EMPTY.");
        else
                preorder(root);

        break;
case 4: if(root==NULL)
                printf("\nTREE IS EMPTY.");
```

```
                        else
                                postorder(root);
                        break;
                default: printf("\nCase not present.");
        }
        printf("\nWANT TO CONTINUE?(Y/N)  ");
        fflush(stdin);
        scanf("%c",&ca);
        }while(ca=='y'||ca=='Y');
getch();
}
```