

Vaidehi M Godbole
Roll no 14

Exp no 6

Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<malloc.h>
struct node{
    int data;
    struct node *left;
    struct node *right;
};
struct node *tree;
void create(struct node *);
struct node *insert(struct node *,int);
void inorder(struct node *);
void preorder(struct node *);
void postorder(struct node *);
void main(){
    int choice,x;
    create(tree);
    do{
        printf("Menu:\t1.Insert a node\t2.Display an inorder
traversal\t3.Display a preorder traversal\t4.Display a postorder
traversal\t5.Exit\nEnter operation to perform:");
        scanf("%d",&choice);
        switch(choice){
            case 1: printf("Enter data to be inserted:");
                    scanf("%d",&x);
                    tree = insert(tree,x);
                    break;
            case 2: printf("Elements in inorder traversal are:");
                    inorder(tree);
                    printf("\n");
                    break;
```

```

        case 3: printf("Elements in preorder traversal are:");
                preorder(tree);
                printf("\n");
                break;
        case 4: printf("Elements in postorder traversal are:");
                postorder(tree);
                printf("\n");
                break;
        case 5: printf("Exiting program...");
                break;
        default: printf("Invalid input!");
    }
}while(choice!=5);
}
void create(struct node *tree){
    tree = NULL;
}
struct node *insert(struct node *tree,int x){
    struct node *p,*temp,*root;
    p = (struct node *) malloc (sizeof(struct node));
    p->data = x;
    p->left = NULL;
    p->right = NULL;
    if(tree == NULL){
        tree = p;
        tree->left = NULL;
        tree->right = NULL;
    }else{
        root = NULL;
        temp = tree;
        while(temp != NULL){
            root = temp;
            if(x<temp->data){
                temp = temp->left;
            }else{
                temp = temp->right;
            }
        }
        if(x<root->data){
            root->left = p;
        }else{
            root->right = p;
        }
    }
}

```

```

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

```

Output:

```
Activities Terminal Sep 1 12:07 dl0417@itadmin: ~
dl0417@itadmin:~$ gcc valdehl.c
dl0417@itadmin:~$ ./a.out
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:1
Enter data to be inserted:12
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:1
Enter data to be inserted:43
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:1
Enter data to be inserted:65
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:1
Enter data to be inserted:42
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:1
Enter data to be inserted:9
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:1
Enter data to be inserted:35
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:1
Enter data to be inserted:23
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:1
Enter data to be inserted:78
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:2
Elements in inorder traversal are:9 12 23 35 42 43 65 78
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:3
Elements in preorder traversal are:12 9 43 42 35 23 65 78
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:4
Elements in postorder traversal are:9 23 35 42 78 65 43 12
Menu: 1.Insert a node 2.Display an inorder traversal 3.Display a preorder traversal 4.Display a postorder traversal 5.Exit
Enter operation to perform:
```