

```

1 #include<stdio.h>
2 #include<conio.h>
3 #include<malloc.h>
4 #include<process.h>
5 struct node
6 {
7     int info;
8     struct node *rlink;
9     struct node *llink;
10 };
11 typedef struct node *NODE;
12 NODE getnode()
13 {
14     NODE x;
15     x=(NODE)malloc(sizeof(struct node));
16     if(x==NULL)
17     {
18         printf("Memory Full\n");
19         exit(0);
20     }
21     return x;
22 }
23 void freenode(NODE x)
24 {
25     free(x);
26 }
27 NODE insert(NODE root,int item)
28 {
29     NODE temp,cur,prev;
30     temp=getnode();
31     temp->rlink=NULL;
32     temp->llink=NULL;
33     temp->info=item;
34     if(root==NULL)
35         return temp;
36     prev=NULL;
37     cur=root;
38     while(cur!=NULL)

```

```

36 prev=NULL;
37 cur=root;
38 while (cur!=NULL)
39 {
40     prev=cur;
41     cur=(item<cur->info)?cur->llink:cur->rlink;
42 }
43 if (item<prev->info)
44     prev->llink=temp;
45 else
46     prev->rlink=temp;
47 return root;
48 }
49 void display(NODE root,int i)
50 {
51     int j;
52     if (root!=NULL)
53     {
54         display(root->rlink,i+1);
55         for (j=0;j<i;j++)
56             printf(" ");
57         printf("%d\n",root->info);
58         display(root->llink,i+1);
59     }
60 }
61 NODE delete(NODE root,int item)
62 {
63     NODE cur,parent,q,suc;
64     if (root==NULL)
65     {
66         printf("empty\n");
67         return root;
68     }
69     parent=NULL;
70     cur=root;
71     while (cur!=NULL&&item!=cur->info)
72     {
73         parent=cur;

```

```

71 while (cur!=NULL&&item!=cur->info)
72 {
73     parent=cur;
74     cur=(item<cur->info)?cur->llink:cur->rlink;
75 }
76 if (cur==NULL)
77 {
78     printf("not found\n");
79     return root;
80 }
81 if (cur->llink==NULL)
82     q=cur->rlink;
83 else if (cur->rlink==NULL)
84     q=cur->llink;
85 else
86 {
87     suc=cur->rlink;
88     while (suc->llink!=NULL)
89         suc=suc->llink;
90     suc->llink=cur->llink;
91     q=cur->rlink;
92 }
93 if (parent==NULL)
94     return q;
95 if (cur==parent->llink)
96     parent->llink=q;
97 else
98     parent->rlink=q;
99 freenode (cur);
100 return root;
101 }
102
103 void preorder (NODE root)
104 {
105     if (root!=NULL)
106     {
107         printf ("%d\n", root->info);
108         preorder (root->llink);

```

```

106     {
107         printf("%d\n", root->info);
108         preorder(root->llink);
109         preorder(root->rlink);
110     }
111 }
112 void postorder(NODE root)
113 {
114     if(root!=NULL)
115     {
116
117         postorder(root->llink);
118         postorder(root->rlink);
119         printf("%d\n", root->info);
120     }
121 }
122 void inorder(NODE root)
123 {
124     if(root!=NULL)
125     {
126
127         inorder(root->llink);
128         printf("%d\n", root->info);
129         inorder(root->rlink);
130     }
131 }
132 int main()
133 {
134     int item, choice;
135     NODE root=NULL;
136     for(;;)
137     {
138         printf("\n1.Insert\n2.Display\n3.Preorder\n4.Postorder\n5.Inorder\n6.Delete\n7.Exit\n");
139         printf("Enter the Choice :");
140         scanf("%d", &choice);
141         switch(choice)
142         {
143             case 1:printf("Enter the Item :\n");

```

```

128     printf("%d\n", root->info);
129     inorder(root->rlink);
130 }
131 }
132 int main()
133 {
134     int item, choice;
135     NODE root=NULL;
136     for(;;)
137     {
138         printf("\n1.Insert\n2.Display\n3.Preorder\n4.Postorder\n5.Inorder\n6.Delete\n7.Exit\n");
139         printf("Enter the Choice :");
140         scanf("%d", &choice);
141         switch(choice)
142         {
143             case 1:printf("Enter the Item :\n");
144                     scanf("%d", &item);
145                     root=insert(root, item);
146                     break;
147             case 2:display(root, 0);
148                     break;
149             case 3:preorder(root);
150                     break;
151             case 4:postorder(root);
152                     break;
153             case 5:inorder(root);
154                     break;
155             case 6:printf("Enter the Item :\n");
156                     scanf("%d", &item);
157                     root=delete(root, item);
158                     break;
159             default:exit(0);
160                     break;
161         }
162     }
163     return 0;
164 }
165

```

```
1.Insert
2.Display
3.Preorder
4.Postorder
5.Inorder
6.Delete
7.Exit
Enter the Choice :1
Enter the Item :
100
```

```
1.Insert
2.Display
3.Preorder
4.Postorder
5.Inorder
6.Delete
7.Exit
Enter the Choice :1
Enter the Item :
20
```

```
1.Insert
2.Display
3.Preorder
4.Postorder
5.Inorder
6.Delete
7.Exit
Enter the Choice :1
Enter the Item :
10
```

```
1.Insert
2.Display
3.Preorder
4.Postorder
5.Inorder
6.Delete
7.Exit
Enter the Choice :1
```

6.Delete

7.Exit

Enter the Choice :1

Enter the Item :

30

1.Insert

2.Display

3.Preorder

4.Postorder

5.Inorder

6.Delete

7.Exit

Enter the Choice :2

100

30

20

10

1.Insert

2.Display

3.Preorder

4.Postorder

5.Inorder

6.Delete

7.Exit

Enter the Choice :1

Enter the Item :

200

1.Insert

2.Display

3.Preorder

4.Postorder

5.Inorder

6.Delete

7.Exit

Enter the Choice :1

Enter the Item :

150

1.Insert

150

- 1.Insert
- 2.Display
- 3.Preorder
- 4.Postorder
- 5.Inorder
- 6.Delete
- 7.Exit

Enter the Choice :1

Enter the Item :

300

- 1.Insert
- 2.Display
- 3.Preorder
- 4.Postorder
- 5.Inorder
- 6.Delete
- 7.Exit

Enter the Choice :2

300

200

150

100

30

20

10

- 1.Insert
- 2.Display
- 3.Preorder
- 4.Postorder
- 5.Inorder
- 6.Delete
- 7.Exit

Enter the Choice :3

100

20

10

30

200

150

300

1.Insert

2.Display

3.Preorder

4.Postorder

5.Inorder

6.Delete

7.Exit

Enter the Choice :4

10

30

20

150

300

200

100

1.Insert

2.Display

3.Preorder

4.Postorder

5.Inorder

6.Delete

7.Exit

Enter the Choice :5

10

20

30

100

150

200

300

1.Insert

2.Display

3.Preorder

4.Postorder

5.Inorder

6.Delete

7.Exit

4.Postorder

5.Inorder

6.Delete

7.Exit

Enter the Choice :4

10

30

20

150

300

200

100

1.Insert

2.Display

3.Preorder

4.Postorder

5.Inorder

6.Delete

7.Exit

Enter the Choice :5

10

20

30

100

150

200

300

1.Insert

2.Display

3.Preorder

4.Postorder

5.Inorder

6.Delete

7.Exit

Enter the Choice :7

(program exited with code: 0)