Lab-Program-10

```
#include <stdio.h>
#include <malloc.h>
#include <Process.h>

struct node
{
 int info;
 struct node *rlink;
 struct node *llink;
};
typedef struct node *NODE;
NODE getnode()
{
 NODE x;
 x=(NODE) malloc (sizeof(struct node));
 if(x==NULL)
 {
  Printf("Memory Full\n");
  exit(0);
 }
 return x;
}
void freenode(NODE x)
{
 free(x);
}
```

```
NODE insert(NODE root, int item)
{
 NODE temp, cur, Prev;
 temp = getnode();
 temp->rlink=NULL;
 temp->llink=NULL;
 temp->info=item;
 if(root==NULL)
 return temp;
 Prev=NULL;
 cur=root;
 while(cur!=NULL)
 {
  Prev=cur;
  cur=(item<cur->info)?cur->llink:cur->rlink
 }
 if(item<Prev->info)
 Prev->llink=temp;
 else
 Prev->rlink=temp;
 return root;
}
void display(NODE root, int i)
{
 int j;
 if(root!=NULL)
 {
  display(root->rlink, i+1);
```

```
    for(j=0;j<i;j++)
      Printf(" ");
      Printf("%d\n", root->info);
      display(root->llink, i+1);
    }
}

NODE delete(NODE root, int item)
{
  NODE cur, Parent, q, suc;
  if (root==NULL)
  {
    Printf("empty\n");
    return root;
  }
  Parent=NULL;
  cur = root;
  while (cur!=NULL && item!=cur->info)
  {
    Parent=cur;
    cur=(item<cur->info)? cur->llink:cur->rlink;
  }
  if (cur==NULL)
  {
    Printf("not found\n");
    return root;
  }
```

```
if (cur->llink==NULL)
  q=cur->rlink;
else if (cur->rlink==NULL)
  q=cur->llink;
else
{
  suc=cur->rlink;
  while (suc->llink!=NULL)
  suc=suc->llink;
  suc->llink=cur->llink;
  q=cur->rlink;
}
if (Parent==NULL)
return q;
if (cur==Parent->llink)
Parent->llink=q;
else
  Parent->rlink=q;
  freenode(cur);
  return root;
}
void Preorder(NODE root)
{
  if(root!=NULL)
  {
    Postorder
    Printf("%d\n", root->info);
```

```
    Preorder(root->llink);
    Preorder(root->rlink);
  }
}
void Postorder(NODE root)
{
  if(root != NULL)
  {
    Postorder(root->llink);
    Postorder(root->rlink);
    Printf("%d\n", root->info);
  }
}
void inorder(NODE root)
{
  if(root != NULL)
  {
    inorder(root->llink);
    Printf("%d\n", root->info);
    inorder(root->rlink);
  }
}
int main()
{
  int item, choice;
  NODE root = NULL;
```

```
for(;;)
{
Printf("\n1.Insert \n2.Display \n3.Preorder \n
    \n4.Inorder \n5.Delete \n 7.Exit \n");
Printf("Enter the choice : ");
Scanf("%d",&choice);
switch(choice)
{
 case 1 : Printf("Enter the Item :\n");
          Scanf("%d", &item);
          root= insert(root, item);
          break;

 case 2 : display(root,0);
          break;
 case 3 : Preorder(root);
          break;
 case 4 : Postorder(root);
          break;
 case 5 : inorder(root);
          break;
 case 6 : Printf("Enter the Item :\n");
          Scanf("%d", &item);
          root =delete(root, item);
          break;
 default : exit(0);
          break;
 }
}return 0;
```