```
#include<stdio.h>
#include<stdlib.h>
struct node
{ int value;
struct node *l;
struct node *r;
};
void preorder(struct node* root)
{ if (root == NULL)
return;
printf("%d ", root->value);
preorder (root->l);
preorder (root->r);
}
void postorder(struct node* root)
{ if (root == NULL) return;
postorder(root->l);
postorder (root->r);
printf( "%d ", root->value);
}
void inorder(struct node* root)
{ if (root == NULL) return;
inorder (root->l);
printf("%d ", root->value);
inorder (root->r);
}
struct node *f_Node(int value)
{
struct node *new;
```

```
new = malloc(sizeof (struct node));
new->value = value;
new->l= NULL;
new->r = NULL;
return new;
}
struct node *insert(struct node *root, int value)
{
if (root == NULL) return f_Node(value);
if(root->value < value)
root->r = insert(root->r, value) ;
else if (root->value > value)
root->l = insert(root->l, value);
return root;
}

int main()
{
struct node *root = NULL;
int temp_n;
int ele;
printf("total elements:");
scanf( "%d", &ele);
for (int i =0; i<ele;i++)
{
printf("enter number:");
scanf( "%d", &temp_n);
root = insert(root, temp_n);
}
```

```
printf("preorder:\n");
preorder (root);
printf("\nInorder : \n");
inorder (root);
printf("\npostorder :\n");
postorder (root);

return 0;
}
```

Output:

```
total elements:4
enter number:10
enter number:40
enter number:32
enter number:90
preorder:
10 40 32 90
Inorder:
10 32 40 90
postorder:
32 90 40 10
```