ANVITHA
1BM19CS021

```
#include <stdio.h>
#include <stdlib.h>
struct node
{
  int data;
  struct node * left;
  struct node * right;
} *root1;

struct node * create()
{
struct node * temp;
printf("\n Enter data:");
temp = (struct node *) malloc (sizeof (struct node
scanf = ("%d", &temp->data);
temp -> left = temp -> right = NULL;
 return temp;
}
void insert (struct node* root, struct node *temp)
{
if (temp-> data < root->data)
{
if (root -> left != NULL)
insert (root-> left, temp);
else
root -> left = temp;
}
if (temp -> data > root -> data)
{
if (root -> right != NULL)
insert (root -> right, temp);
else
root -> right = temp;
}
}
```

```
void Postorder (struct node * node)
{
  if (node == NULL)
    return;
  Postorder (node -> left);
  Postorder (node -> right);
  print("%d", node -> data);
}

void Inorder (struct node *node)
{
  if (node == NULL)
    return;
  Inorder (node -> left);
  printf ("%d", node -> data);
  Inorder (node -> right);
}

void Preorder (struct node * node)
 {
  if (node == NULL)
    return;
  printf ("%d", node -> data);
  Preorder (node -> left);
  Preorder (node -> right);
}
int main()
 {
  int ch;
  struct node *temp;
  do
   {
    printf ("1. create 2. Insert 3. preorder 4. postorder
                                    5. inorder
                                    6. Exit ");
```

```
scanf("%d", &ch);
switch (ch)
{
case 1:
root1 = create();
break;
case 2:
printf("Enter the elem to be created \n");
temp = (struct node *) malloc (sizeof (struc
scanf("%d", &temp -> data);
insert (root1, temp);
break;
case 3:
Preorder(root1);
break;
case 4:
Postorder (root1);
break;
case 5:
Inorder (root1);
break;
case 6:
break;
default:
printf("Wrong choice");
}
} while (ch != 6);
}
```