LAB PROGRAM-10

1BM19CS052

```
#include <stdio.h>
#include <stdlib.h>
typedef struct Node {
  int data;
struct Node *left, *right;
} node;
node *create (int data) {
  node *temp;
  temp = (node *) malloc (sizeof(node));
  temp -> data = data;
  temp -> left = temp -> right = NULL;
 return temp;
}
 void inorder(node *root) {
  if (root != NULL) {
  inorder (root -> left);
  printf ("%d", root -> data);
   inorder (root -> right);
  }
 }
 void preorder (node *root) {
  if (root != NULL) {
   printf ("%d", root -> data);
   preorder (root -> left);
   preorder (root -> right);
  }
 }
 void postorder ( node *root) {
  if (root != NULL) {
```

```
postorder (root -> left );
 postorder ( root -> right);
 printf ( "%d", root -> data );
 }
}
void insert (node *root, 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;
   }
 }
int main (void) {
 node * root = NULL, * temp;
   int  choice =0;
   while (choice != 2)
   {
     temp =
   printf ("1- Insert\n");
   printf ("2- Exit \n");
   printf ("Enter your choice : ");
    scanf ("%d", &choice );
    if (choice == 1)
    {
```

```
int val;
printf ("Enter value : ");
scanf ("%d", &~~choice~~ val);
temp = create (val);
 if (root == NULL)
    root = temp;
  else
    insert (root, temp);
 }
  else if (choice == 2)
     break;
  else
   printf ("Invalid choice\n");
 }
printf ("Inorder traversal: ");
inorder (root);
printf ("\n Preorder traversal:");
preorder (root);
printf ("\n Post order traversal :");
postorder (root);
}
```