LAB PROGRAM - 8

Q. write a program

a) To construct binary search tree.

b) To traverse the tree using all the methods
i.e, inorder, pre-order and post order.

c) To display the elements in the tree.

code :

```
#include <stdio.h>
# include <stdlib.h>

struct node {
        int data;
        struct node *left, *right;
};
```

```
struct Node *createNode (int value)
{
 struct Node * newNode = (struct Node *)
 malloc (size of (struct Node ));
 new Node -> data = value;
 newnode ->left = newnode -> right = NULL.
 return newNode ;
}.
struct Node * insert (struct Node * root,
     int value)
 if (root == NULL )
    return createNode (value );

  if (value < root -> data)
   root ->left = insert (root ->left , value);

  else if (value > root -> data ),
   root -> right = insert (root -> right,
                    value);
    return root;

void inorder (struct Node *root )
{
 if (root == NULL )
   return;
  inorder (root ->left );
    printf (" %d ", root -> data );
     inorder (root -> right );
 }.

void preorder (struct node * root )
 {
  if ( root == NULL ) return;
```

```
printf ("%d ", root->data);
preorder (root->left);
preorder (root->right);

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

void display (struct node * root)
{
  printf ("BST elements (Inorder):");
  inorder (root).
  printf ("\n");
}.

int main ()
 struct node *root = NULL;
 int choice, value;
 while (1)
 {
 printf ("\n --- Binary search Tree
          menu ---\n").
printf ("1. Insert into BST \n");
printf ("2. Inorder Traversal \n");
printf ("3. Preorder Traversal \n");
printf ("4. Postorder Traversal \n");
printf ("5. Display BST \n")
printf ("6. exit \n");
printf ("Enter choice:");
```

```
scanf ("%d", &choice );
switch (choice)
{
case 1 :
     printf ("Enter value to insert:");
     scanf ("%d", &value );
     root = insert (root ,value );
     break ;

case 2 :
      printf (" Inorder Traversal:");
       preorder (root);
       printf ("\n");
        break ;

case 3 :
     printf (" preorder Traversal:");
      preorder (root);
      printf ("\n");
      break;

case 4:
    printf (" postorder Traversal:");
    postorder (root);
    printf ("\n");
    break;

case 5:
       display (root);
       break;

case 6 :
       exit (0);
default:
    printf("Invalid choice: Try again
                  .\n");

    }
  }
     return 0;
  }.
```

output!

```
--- Binary search Tree Menu ---
1. Insert into BST
2. Inorder Traversal
3. preorder Traversal
4. postorder Traversal
5. Display BST
6 Exit
   Enter choice: 1
   enter value to insert: 50

--- Binary search Tree Menu ---
1. Insert into BST
2. Inorder Traversal
3. preorder Traversal
4. postorder Traversal
5- Display BST
6. Exit
Enter choice: 1
Enter value to insert: 30
```