Name : Ramya Ramesh
USN : 1BM19CH038

Lab 9 : Binary Search Tree Traversal

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

struct node {
  int value;
  struct node *left;
  struct node * right;
 } *root = NULL, *temp = NULL, *t2, *t1;
void insert();
void inorder(struct node *t);

void search(struct node *t);
void preorder (struct node *t);
void postorder (struct node *t);

void main () {
    int ch;
    while (1)
    {
        printf ("\n ***MENU***\n);
        printf ("1. Insert an element into tree\n");
        printf ("2. Inorder Traversal \n");
        printf ("3. Preorder Traversal\n");
        printf ("4. Postorder Traversal\n");
        printf ("5. Exit\n");
        printf ("\n Enter your choice : ");
        scanf ("%d", &ch);
        switch(ch) {
```

```
case 1 : insert() ;
         break ;
case 2 : inorder (root);
         break ;
case 3 :  preorder (root);
           break;
case 4 :  post order(root);
           break;
case 5 : exit(0);
default : printf (" \n Invalid Choice!");
             break;
     }
   }
}
void insert() {
    int data;
    printf(" Enter data to be inserted :  ");
     scanf (" %d" , &data);
    temp = (struct node *)malloc( sizeof (struct node));
    temp → value = data;
    temp → left = temp → right = NULL;

          if (root == NULL)
              root = temp;
          else
              Search (root) ;
    }
```

Page No.:
Date:
YOUVA

```
void search(struct node *t) {
    if((temp→value > t→value) && (t→right != NULL))
      search(t→right)
    else if((temp→value> t→value) && (t→right ==NULL))
      t→right = temp;
    else if ((temp→ value <t→value) && (t→left != NULL))
    search(t→left)
     else if ((temp→value <t→value) && (t→left ==NULL))
        t→left = temp;
}

void inorder (struct node *t)
{
      if (root == NULL)
      {
          printf (" No elements in the tree\n");
        return;
      }
   if (t→left != NULL)
        inorder (t→left);
    printf ("%d →", t→value);
    if(t→right != NULL)
     inorder(t→right);
  }

void    preorder(struct node *t)
        {
            if (root == NULL) {
                printf("No elements in the tree !\n");
                return;
            } printf ("%d →  ", t→value);
   if (t→ left != NULL)
              preorder (t→left);
   if ( t→right != NULL)
             preorder (t→right);
   }
```

Date:

YOUVA

```
void postorder (struct node *t)
      {
            if (root == NULL)
            {
                  printf ("No elements in the tree! \n");
                  return;
            }
      if (t→left != NULL)
            postorder (t→left);
      if (t→ right != NULL)
            postorder (t→right);
      printf ("%d→   ", t→value);
}
```