/*search in binary tree*/

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

//Represent a node of binary tree
struct node{
   int data;
   struct node *left;
   struct node *right;
};

//Represent the root of binary tree
struct node *root = NULL;

static bool flag = false;

//createNode() will create a new node
struct node* createNode(int data){
   //Create a new node
   struct node *newNode = (struct node*)malloc(sizeof(struct node));
   newNode->data = data;
   newNode->left = NULL;
   newNode->right = NULL;

   return newNode;
}

void searchNode(struct node *temp, int value){
   //Check whether tree is empty
   if(root == NULL){
```

```
        printf("Tree is empty\n");
    }
    else{

        if(temp->data == value){
            flag = true;
             return;
        }

        if(flag == false && temp->left != NULL){
        searchNode(temp->left, value);
        }

        if(flag == false && temp->right != NULL){
        searchNode(temp->right, value);
        }
    }
}

int main()
{
    //Add nodes to the binary tree
    root = createNode(1);
    root->left = createNode(2);
    root->right = createNode(3);
    root->left->left = createNode(4);
    root->right->left = createNode(5);
    root->right->right = createNode(6);

      searchNode(root, 5);
```

```
    if(flag)
        printf("Element is present in the binary tree");
    else
        printf("Element is not present in the binary tree");
    return 0;
}
```