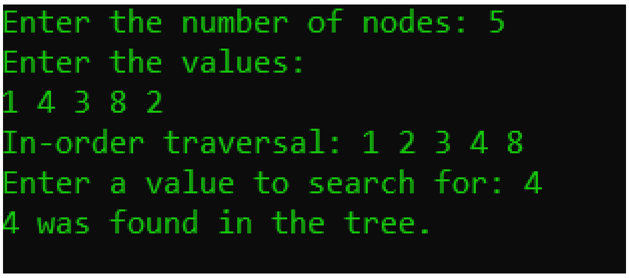
Program - 13

Title : **Write a Program to Implement the Binary Search Tree Operations.**

Code:

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

// Define the tree node structure
struct node {
    int data;
    struct node* left;
    struct node* right;
};

// Function to create a new node
struct node* createNode(int val) {
    struct node* newNode = (struct node*) malloc(sizeof(struct node));
    newNode->data = val;
    newNode->left = NULL;
    newNode->right = NULL;
    return newNode;
}

// Function to insert a value into the tree
struct node* insert(struct node* root, int val) {
    if (root == NULL) {
        return createNode(val);
    } else if (val < root->data) {
        root->left = insert(root->left, val);
    } else {
        root->right = insert(root->right, val);
    }
    return root;
}

// Function to search for a value in the tree
struct node* search(struct node* root, int val) {
    if (root == NULL || root->data == val) {
        return root;
    } else if (val < root->data) {
        return search(root->left, val);
    } else {
        return search(root->right, val);
    }
}

// Function to print the tree in-order
void inOrder(struct node* root) {
    if (root != NULL) {
        inOrder(root->left);
        printf("%d ", root->data);
        inOrder(root->right);
    }
}

// Main function to test the tree implementation
int main() {
    struct node* root = NULL;
    int n, val;
    printf("Enter the number of nodes: ");
    scanf("%d", &n);
    printf("Enter the values:\n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &val);
        root = insert(root, val);
    }
    printf("In-order traversal: ");
    inOrder(root);
    printf("\n");
    printf("Enter a value to search for: ");
    scanf("%d", &val);
    struct node* result = search(root, val);
    if (result == NULL) {
        printf("%d was not found in the tree.\n", val);
    } else {
        printf("%d was found in the tree.\n", val);
    }
    return 0;
}
```

Output:



```
Enter the number of nodes: 5
Enter the values:
1 4 3 8 2
In-order traversal: 1 2 3 4 8
Enter a value to search for: 4
4 was found in the tree.
```

Date : ___/____/______

Teacher Sign ………………………………………………………