**Name: Zeeshan Ali (SU92-BSITM-F22-019)**

**Data Structure & Algorithm(Lab)**

**Lab # 08 Task**

**Question:**

Binary Search Tree Simple program to create a BST of integers and search an

element in it. 15 root node

/ \

10 20

/ \ / \

8 12 17 25

**Code:**

#include<iostream>

using namespace std;

struct BstNode {

int data;

BstNode\* left;

BstNode\* right;

};

BstNode\* GetNewNode(int data) {

BstNode\* newNode = new BstNode();

newNode->data = data;

newNode->left = newNode->right = NULL;

return newNode;

}

BstNode\* Insert(BstNode\* root,int data) {

if(root == NULL) {

root = GetNewNode(data);

}

else if(data <= root->data) {

root->left = Insert(root->left,data);

}

else {

root->right = Insert(root->right,data);

}

return root;

}

bool Search(BstNode\* root,int data) {

if(root == NULL) {

return false;

}

else if(root->data == data) {

return true;

}

else if(data <= root->data) {

return Search(root->left,data);

}

else {

return Search(root->right,data);

}

}

int main() {

BstNode\* root = NULL;

root = Insert(root,15);

root = Insert(root,10);

root = Insert(root,20);

root = Insert(root,25);

root = Insert(root,8);

root = Insert(root,12);

root = Insert(root,17);

int number;

cout<<"Enter number be searched\n";

cin>>number;

if(Search(root,number) == true)

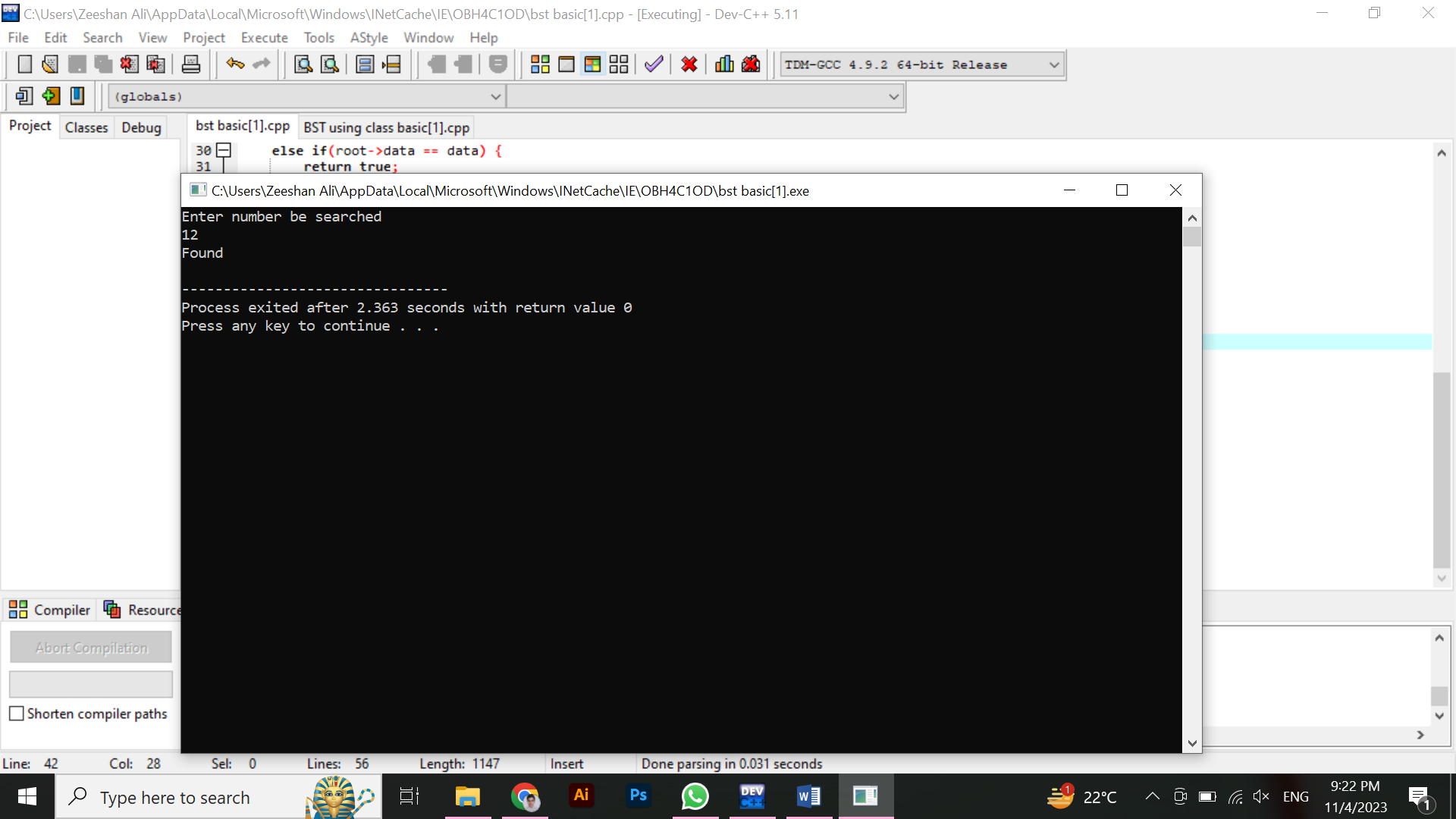
cout<<"Found\n";

else

cout<<"Not Found\n";

}

**Output:**

****