WORKSHEET 4

Student Name: Avinash Kumar UID:22BCS10229

Branch: CSE Section/Group:603-B

Semester: 3rd Date of Performance:31/10/2023 Subject Name: Data Structures Subject Code: 22CSH211/22ITH211

1. **Aim**:

Write Function to Find minimum and maximum value in a binary search tree transverse the tree accordingly to find the nodes with minimum and maximum value.

1. **Source Code:**

#include <iostream> using namespace std; class TreeNode { public:

int value; TreeNode\* left; TreeNode\* right;

TreeNode(int val) : value(val), left(nullptr), right(nullptr) {}

**};**

int findMin(TreeNode\* node) { if (node == nullptr) {

return -1;

**}**

while (node->left) { node = node->left;

**}**

return node->value;

**}**

int findMax(TreeNode\* node) { if (node == nullptr) {

return -1;

**}**

while (node->right) { node = node->right;

**}**

return node->value;

**}**

int main() {

TreeNode\* root = new TreeNode(10); root->left = new TreeNode(5);

root->right = new TreeNode(15); root->left->left = new TreeNode(3); root->left->right = new TreeNode(7);

root->right->left = new TreeNode(12); root->right->right = new TreeNode(28);

int minValue = findMin(root); int maxValue = findMax(root);

if (minValue != -1) {

cout << "Minimum value in the BST: " << minValue <<endl;

} else {

cout << "The tree is empty." << endl;

**}**

if (maxValue != -1) {

cout << "Maximum value in the BST: " << maxValue << endl;

} else {

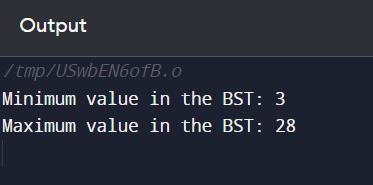
cout << "The tree is empty." << endl;

**}**

return 0;

**}**}

1. **Screenshot of Outputs:**



1. **Learning Outcomes**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1: Understanding of Binary Search Trees (BSTs)** | | | | |
| **2:** | **Data Structure Familiarity** | | |  |
| **3:About Tree Transversal** | | |  | |
| **4:Algorithm complexity** | |  | | |