**Output Lab 8**

**Bibek Dhungana**

**Case 1: with input given in the lab.**

BinaryTreeNode\* root = new BinaryTreeNode(4);

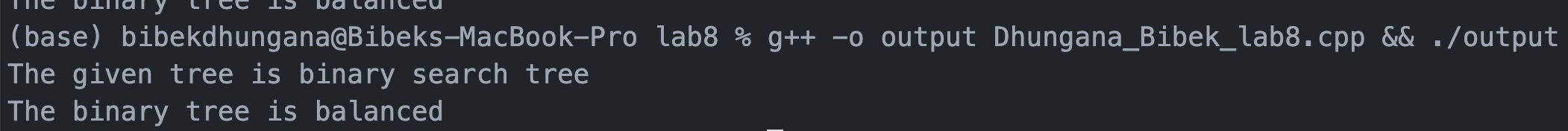
root->left = new BinaryTreeNode(2);

root->right = new BinaryTreeNode(5);

root->left->left = new BinaryTreeNode(1);

root->left->right = new BinaryTreeNode(3)

The tree is tree is balanced and is also binary search tree.



**Case 2: Balanced but not Binary search tree**

BinaryTreeNode\* root = new BinaryTreeNode(20);

root->left = new BinaryTreeNode(2);

root->right = new BinaryTreeNode(5);

root->left->left = new BinaryTreeNode(1);

root->left->right = new BinaryTreeNode(3)

**Text

Description automatically generated with medium confidence**

**Case 3: Binary search tree but not balanced**

BinaryTreeNode\* root = new BinaryTreeNode(4);

root->left = new BinaryTreeNode(2);

root->right = new BinaryTreeNode(5);

root->left->left = new BinaryTreeNode(1);

root->left->right = new BinaryTreeNode(3)

root->left->left->left = new BinaryTreeNode(0);

Graphical user interface

Description automatically generated with medium confidence

**Case 4: Not Binary search tree and not balanced.**

BinaryTreeNode\* root = new BinaryTreeNode(20);

root->left = new BinaryTreeNode(2);

root->right = new BinaryTreeNode(5);

root->left->left = new BinaryTreeNode(1);

root->left->right = new BinaryTreeNode(3)

root->left->left->left = new BinaryTreeNode(0);

**Graphical user interface, text, application

Description automatically generated**