

CSCI 232 Lab 3

Due Wednesday May 24th @ 11:59 PM. Please submit this assignment (.java files) to the appropriate dropbox on D2L

Background and Instructions

In this lab, you will build some methods for a simple binary search tree that stores integers. Use the code we developed in class on May 22nd/23rd (may22.zip) as a starting point

You may not change any code in the **Node** class. Add the following methods (with the exact methods headers given below) to the **BST** class:

Add the following methods (with the exact methods headers given below) to the BST class:

1. **public int getMin()**. This method will return the minimum value in the BST. If there are no values in the BST, return -1.
2. **public int getMax()**. This method will return the maximum value in the BST. If there are no values in the BST, return -1.
3. **public Node find(int value)**. This method will return the Node whose value equals the input parameter value, if in fact that input parameter value exists in the BST. If the BST does not contain the input parameter value, return null. This method may not exhaustively search the whole tree for the input parameter value. It must go from the root to the input parameter value via the shortest path, just as our **insert()** method in class did. You are not allowed to use breadthFirst or depthFirst search.

Test your code to make sure it works how you want it to before you turn it in.

Grading

Grading will be done as follows:

- The getMin() method is correct - 3 points
- The getMax() method is correct - 3 points
- The find() method is correct - 4 points

NOTE: If your code does not compile, correctness cannot be verified and you won't receive any points for your code. Turn in code that compiles!

Hints

- Look at a BST and figure out where the minimum and maximum values will be.

- `find()` will function very similarly to the `insert()` method.