

Programming Assignment #7

Binary Search Tree implementation

CS 3358.501, Summer I 2012

Instructor: Jill Seaman

Due: Thursday, 7/5/2012 (upload electronic copy by 4:30pm)

Problem:

You will implement the BST_3358 interface, which is a templated binary search tree.

[bst_3358.h](#) the interface and incomplete implementation

[bst_test.cpp](#) a test file

[bst_out.txt](#) the expected output for the test file

In the file `bst_3358.h`, you will find 10 functions with **//implement me** in their function definition bodies. These are the ones you need to implement.

Note that there are 15 public functions. There are 12 private functions. Many of the public functions call a private function to do their work. The private functions take `TreeNode`s as a parameter so they may be called recursively (the public functions cannot). You will implement 10 of the private functions:

- `copyTree`
- `findItem`
- `findMin`
- `findMax`
- `insertItem`
- `countNodes`
- `makeEmpty`
- `inOrderTraversal`
- `preOrderTraversal`
- `postOrderTraversal`

All of the other functions (including `delete`) have been implemented.

NOTES:

- `bst_3358.h` is set up so it will already compile with the `bst_test.cpp` file. Implement one function at a time, compile, test, and fix.

- I recommend implementing insert, and then find and inOrderTraversal, so that you can test your insert. Everything depends on getting insert implemented correctly.
 - Most of the definitions will be short, one or two recursive calls with a base case or two. Don't try to make it more complicated than it needs to be. Don't say "it can't be that simple". It could be.
-

Style:

See the Style Guidelines document on the course website.

Logistics:

Please submit the following file:

`bst_3358_XXXXXX.h`

The xxxxx is your TX State NetID (your txstate.edu email id).

Submit: an electronic copy only, using the Assignments tool on the TRACS website for this class.