Ass 2: Binary Search Tree

[**updated 10/8 to include instruction on zip file**]

Implement a BinarySearchTree based on the template below. BinarySearchTree must be able to handle different types of data. the provided sample ass2.cpp tests it using string and int

BinaryNode is already provided and should not be modified.

All the public functions of BinarySearchTree must be implemented

binarysearchtree.h

binarysearchtree.cpp – stub file to help compile

BinaryNode – use as provided. Do not modify. No need to comment further

binarynode.h

binarynode.cpp - included via binarynode.h, cannot be compiled independently

Sample ass2.cpp file, expand it to suit your needs. Automated test will not run your ass2.cpp since it requires keyboard interaction.

ass2.cpp

As always expected when programming, comment clearly and thoroughly. Clearly state any assumptions you make in the beginning comment block of the appropriate place, e.g., the class definition. Comments in the class definition file should describe the ADT, all functionality, and assumptions so someone could use the class and understand behavior and restrictions. Pre and post conditions are fine, but not required. See the example on Assignments page for a well-documented program.

You do NOT need to handle data type errors due to bad input.

I will run my own main to test your code. The main function provided doesn't test your program fully, so you need to supplement it.

Write one function at a time. Test it before moving on to the next function. I suggest starting with add Use valgrind to check for memory leaks as you develop the program. Much easier to fix things early on.

Submit a single zip file, ass2.zip with the following files in it. When unzipped ass2.zip must create a directory named ass2 and place all the files in that directory. See http://faculty.washington.edu/pisan/cpp/creating-zip.html on how to create a zip file.

Class names start with capital letters, but file names are all lowercase for compatibility

binarysearchtree.h

binarysearchtree.cpp

ass2.cpp – your own testing functions and main

output.txt - the script file, as defined in Connecting and compiling files on linux labs

You do not need to submit

binarynode.h

binarynode.cpp

These files should not be modified, so they must be as given in assignment.

Once your code is working on your own machine, test it once more on the linux machines (you have been testing incrementally and using valgrind, right?). See Connecting and compiling files on linux labs

Under unix, compile your code using

```
g++ -std=c++14 -g -Wall -Wextra ass2.cpp -o ass2
```

and create the output.txt file following the instructions on that page.

Submission

Your submission must be named ass2.zip. Your submission must create a directory called ass2 and place all the files in that directory when it is unzipped.

DO NOT include unnecessary files, only submit what is required. DO NOT zip up your Debug directory, your project directory, etc.

Canvas will automatically rename your zip file ass2-2.zip, ass2-3.zip, etc depending on how many times you submit it. *This is fine*. Do not change your zip file name.

Grading Rubric

Multiple criteria. -5 for partially correct, -10 for not working or missing

1. Constructors: Empty, 1 parameter and Copy Constructor
2. Destructor, clear
3. isEmpty, getHeight, numberOfNodes, contains
4. operator==, operator!=
5. add
6. inorderTraverse
7. rebalance
8. readTree
8. memory leaks
10. efficiency and complexity
12. Coding style + ass2.zip constructed properly