Due: Mar 16, Wednesday

In this assignment, you are asked to implement Binary Search Tree and AVL Tree and compare the results (heights) of the two. The two classes are derived classes of a given map class.

1. Make directory ~/IT279/Asg3/ on your Unix account for this assignment, where ~ denotes your home directory. All your works should put directly under this directory before you run the script `copy279all.sh` to copy into the secret directory, otherwise `copy279all.sh` can't find your works. Also, make sure your home directory permission is set to 711.

2. Copy all files from my `/home/cli2/public/IT279/Asg3/` to your own ~/IT279/Asg3/. Among those files, `BST.h` and `AVL.h` are two header files for the two classes `BST<T>` and `AVL<T>`, respectively, that you have to implement. Some details are described as follows. Most of them are self-explained, but if you have any doubt or confusion, don't make assumption, do ask. Or, you can discuss in the discussion group where I will keep an eye on.

   (a) `map.cpp` is the program for the base class `map<T>`. You should not modify this file. All functions defined in this class are virtual, therefore the derived class should define them all.

   (b) `BST.h` is the header file of class `BST<T>`. You should not modify the public section of this class declaration (including the `test()` method). `BST<T>` is a derived class of `map<T>`. The purpose of every method required in the public section is self-explained, except `test()` that doesn't have any specific functionality, and therefore you are free to design it for your testing purpose, but you can't change its prototype.
   The private section is open to you. You can add or modify everything in the private section as helper to your programs. However, if the functions can be static, it's a good programming practice not to show them in the private section.

   (c) `AVL.h` is the header file of class `AVL<T>` that inherits `BST<T>`, where the insert method should be overridden according to the AVL trees rotation principle on insertion. However, you don't have to override the remove method, hence the tree may not be an AVL tree after some removals. If you correctly implement the AVL remove method, you will get extra 10 points.

   (d) `treeNode.cpp` defines a tree node class, which is just a sample implementation of `treeNode<T>` class. You are free to design your own tree node class, but the file name and the class name have to be the same, i.e., `treeNode.cpp` and `TreeNode`. Alternatively, you don't have to maintain a link for the parent, if you can manage to trace the parent node. This in fact may not be a bad idea.

   (e) `Asg3.cpp` is the main program I will simple compile it as `g++ Asg3.cpp`. You should not change any function except `debugging_test()` in `g++ Asg3.cpp`. I keep `debugging_test()` there as an example of how I debug my program. You can modify `debugging_test()` and design your own and uncomment the call in the main function. Together with the public `test()` method in `BST<T>` class, you can check the contents of your tree to see if it constructed correctly. The submitted Asg3.cpp program should comment out `debugging_test()`.

   (f) `Asg3SampleOutputs.txt` is a sample output for your reference. Note that, I set the random number generator with a different seed, therefore the numbers inserted will not be the same. You have to submit your own output.

Secret directory under `public`

Same as before, after you've finished your work, or have decided that what you have is the final version for me to grade, you have to copy all your programs to a secret directory under your `public/IT279myWork` so I can grade them (i.e., compile, run, and check the codes). Select a secret name, say "`peekapoo`" as an example (you should chose your own), to be the name of your secret directory. Then, You can simply copy my `/home/cli2/public/IT279/copy279all.sh` and use the following command line to run it. (You can put `copy279all.sh` in any directory.)

```
bash copy279all.sh peekapoo
```

Regrading the secret name, don't use a name with more than 10 characters, and avoid confusion between letter l and number 1, letter O and number 0. Also note that, Unix's file system is case sensitive.

It is important that all your works for this assignment should be put under the required directory, otherwise `copy279all.sh` can't find your works.

Final Words:

You have to follow the submission guidelines, i.e., cover page (that contains assignment number, student's names, student **ULID**, and secret directory), summary, source code(optional), output, folder, and so on. **No late work will be accepted.**