3460:210 Assignment J

# **Assignment J: The Binary Tree Search**

**Overview**

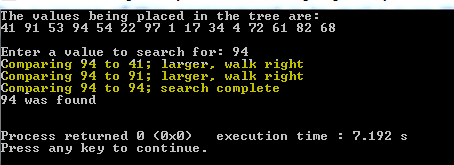
The purpose of this assignment is to make sure that you know how to write a program that uses containers, algorithms, iterators or binary trees.

**PROGRAM SPECIFICATION**

For this assignment, you will complete the binary search tree class by writing the member functions binaryTreeSearch and determineDepth. Search attempts to locate a specified value in a binary search tree object. Depth calculates the depth of the tree. The search function should take as arguments a pointer to the root node of the binary tree and a search key to be located. If the node containing the search key is found, the function should return a pointer to that node; otherwise, the function should return a null pointer. The depth function should take as arguments a pointer to the root node of the binary tree and two pointers, the total depth pointer, and the current depth pointer.

For this assignment you will need to complete the member functions that perform the following tasks:

1. **determineDepth** this member function calculate the depth of the tree.
2. **binarySearchHelper** this member function will perform the binary search on the tree. This function should have display logic exactly as shown below (in yellow).



The specification files have been created for you, along with the driver program as well. The code can be downloaded and is named tree\_main.cpp (to test with), treenode.hpp which contains the TreeNode class-template definition, and the tree.hpp file which contains the Tree class-template definition. This is where you will find the member functions that you are to complete. You need only to update this file and submit that as the final assignment (only submit tree.hpp file). Please use the driver and make sure your program can execute any random test cases successfully.

Make sure that your programs follow good documentation standards and follow the requirements for assignments. Reference the rubric standards on Brightspace.

Submission Instructions – for programming solutions

On Brightspace, go to the matching Assignments for the ASSGN@, where @ is the character of the problem assigned (eg., A for assignment A), and submit the program (cpp) and any (hpp) files.

*Last updated 11.8.2016 by Will Crissey.*

*Be aware that programming falls under all of the rules of plagiarism. Be careful when using any coding found in the outside world that is not your own. Any evidence of plagiarism is subject to sanctions like forfeits, suspension, and even ejection, as determined by the Department of Student Conduct and Community Standards.*