Data Structures (CS 1520) Lab 9 Name:_____

Objectives: You will gain experience BST performance and implementation

To start the lab: Download and unzip the file: http://www.cs.uni.edu/~fienup/cs1520s13/labs/lab9.zip

Part A: Run the `timeBinarySearchTree.py` program that:

- creates a list, evenList, that holds 3,000 sorted, even values (e.g., evenList = [0, 2, 4, 6, 8, ..., 5996, 5998])
- puts (adds) all the evenList items into an initially empty BinarySearchTree object, bst
- times the searches (in) bst for target values 0, 1, 2, 3, 4, ..., 5998, 5999 so half of the searches are successful and half are unsuccessful

a) How long does it take to search for target values from 0, 1, 2, 3, 4, ..., 5998, 5999?

b) Explain why these searches take so long. (Hint: consider the shape of the BinarySearchTree `bst`)

c) Uncomment the "`shuffle(evenList)`" which randomizes the items in evenList. Now how long does it take to search for target values from 0, 1, 2, 3, 4, ..., 5998, 5999?

d) Explain why these searches take so little time.

e) What is the search time with the `timeOpenAddrHashDictSearch.py` program? Why is it faster?

After you have answered the above questions, raise your hand and explain your answers.

Part B:

a) Complete the recursive `height` method in the BinarySearchTree class (model it after one of the traversals). Uncomment the call to the `height` method at the end of the `timeBinarySearchTree.py` program. What is the height of `bst` if we are shuffling the `evenList`?

b) What would be the shortest possible height for a binary tree with 3,000 items?

c) Why was it necessary to raise the recursion limit with the statement: `sys.setrecursionlimit(10000)`?

After you have completed the height method and answered the above questions, raise your hand and explain your answers.