**CSCI 3333**

**Data Structures**

**Spring 2013**

**Final Examination**

**Last Name**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ First Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Student Id: \_\_\_\_\_\_\_

Time allowed: one hour and 45 minutes. Total score: 70 points (10 points for each question.)

This is a ***closed*** book examination.

Answer all questions. Turn in both question and answer sheets (if any). Write your name and student id in the first page of your answer sheets, and your name in every page.

Plan your time well.

**Academic honesty policy will be followed strictly. Cheating will be pursued vigorously and will result in a failing grade of D or below, a permanent academic record and possibly other more serious penalty!**

|  |
| --- |
| **I hereby pledge that I will stay truth to UHCL’s Honor Code.****Signature: Date:**  |

(1) True (T) or False (F)

(a) Any comparison-based sorting algorithm must run in Ω(n log n) time.

(b) Using the definition of the height of an empty tree being -1, the minimum possible height of a binary tree with 128 nodes is 7.

(c) The average time complexity of insertion sort is O(N lg N), where N is the number of elements in the array.

(d) Agile software development is an example of lightweight software development methodology.

(e) The ACM Code of Ethics urges ACM members to avoid harm to others.

(2) Provide a O(N lg N) algorithm, IsSubset(A, B), to check whether every element in an Integer array A appears in another Integer array B. You may assume that elements are unique in both A and B. For examples, if A = {2, 10, 3, 5} and B = {11, 9, 3, 80, 10, 4, 6, {6, 19, 22, 4, 7, 21, 26, 99, 1, 16, 24, 47, 72, 13, 2, 86, 33, 18} 2, 20, 5, 33}, the answer should be true. On the other hand, for A = {10, 222, 3, 5} and B = {11, 9, 3, 80, 10, 4, 6, 21, 30, 5, 33}, the answer should be false since the element 222 of A does not appear in B.

(3) Show the evolution of applying Shell Sort with gap sequence (5, 3, 1) to the array: {6, 19, 22, 4, 7, 21, 26, 99, 1, 16, 24, 47, 72, 13, 2, 86, 33, 18} by showing the array after 5-sorted, 3-sorted and 1-sorted. Show the conceptual view also.

(4)

(a) Convert the following infix expression to the postfix expression: (a+b)\*c/d-e

(b) Convert the following postifx expression to the prefix expression: abcd/\*-ef—

(5) Provide an algorithm for the function CountLarger(p: BST, target: Integer) to count the number of nodes with keys larger than target in a binary search tree p. You may assume that there is a function size(p: BST) that returns the number of nodes in a binary search tree p.

(6)

(a) Show the final BST after the sequential insertions of the following records to an initially empty tree: 4, 80, 23, 16, 99, 5, 1, 2, 70. You may draw a graphical tree or use the bracket notation.

(b) Consider the BST [10 12 [[[[] 16 17] 18 []] 30 77]]. Show the BST after the key 12 is deleted. Use the copy and replace method on the immediate successor if the deleted key resides in an interior node with two children.

(7) Short questions.

(a) Draw the expression tree for the expression: (a-b) \* (c + d) / (e –f)

(b) What is the worst case time complexity of quicksort for an array with N elements?

(c) What is the least number of key comparison of successful search in a binary search tree?