CS340 Winter 1995

Ming Li

CS340 Midterm Exam, Feb. 15, 6-8pm, MC4020/4021

Your Name:
Student ID#:

Question	total marks	your marks
1.	20	
2.	20	
3.	10	
4.	10	
5.	10	
6	5	
7.	15	
8.	10	

Write your name and student number on every page.

Indicate clearly how you have obtained your answers, show all the necessary steps, but be concise.

No books or notes.

If necessary, use the back of the same page to continue with your answer.

Good luck.

1. (20 marks) Answer true or false.

   (a) 8884659 is $O(1)$.

   (b) $\frac{n \log n}{1000000}$ is $O(n)$.

   (c) $100n^2 + n + 3.14159$ is $O(n^2)$.

   (d) $\sqrt{n}$ is $O(n)$.

   (e) If $f(n)$ is $O(n)$ and $g(n)$ is $O(n^2)$, then $f(n) + g(n)$ is $O(n^2)$.

   (f) There is a constant c such that for any AVL tree T, the number of nodes in the left subtree of T is at most c times the number of nodes in the right subtree of T.

   (g) $n^{1000} < 2^n$ for all but finitely many n's.

   (h) The following algorithm runs in $O(n)$ time.

   procedure Puzzle(**integer** n);
   for i **from** 1 **to** n **do**
   for j **from** 1 **to** 10 **do**
   for k **from** n **to** $n + 5$ **do** $x := x + 1$

   (i) For any f, g, either $f(n) = o(g(n))$ or $f(n) = \Omega(g(n))$.

   (j) The time complexity of inserting or deleting an element in an AVL tree of n nodes is $O(n)$.

2. (20 marks) Consider the following tree.

   (a) What is the height of the tree?

   (b) Is this tree a binary search tree?

   (c) Is this tree an AVL tree?

   (d) Is this tree a B-tree of order 3?

   (e) What is the internal path length of this tree?

   (f) What is the external path length of this tree?

   (g) Assuming all elements in the tree are equally likely to be queried, what is the average number of comparisons in a successful search for this tree? In an unsuccessful search (assuming also it is equally likely to end up at any external node)?

   (h) List the nodes of the tree according to inorder traversal.

   (i) List the nodes of the tree according to preorder traversal.

   (j) List the nodes of the tree according to postorder traversal.

3. (10 marks) Given letters A,B,C,D,E,F,G with respective probabilities 0.1 (A), 0.22 (B), 0.23 (C), 0.11 (D), 0.02 (E), 0.03 (F), 0.29 (G). Construct the Huffman code for these letters using Huffman's algorithm. (Draw the tree.)

4. (10 marks) Assume that n is a power of 2.

   (a) $T(1) = 2$; $T(n) = 2 + T(n-1)$. Solve $T(n)$ exactly.

   (b) $T(1) = b$; $T(n) = b + 2T(n/2)$. Solve $T(n)$ exactly.

5. (10 marks) Show the final skip list obtained by inserting the following elements in a skip list that was initially empty. The number of pointers for each element is given in parenthesis (in practice, this would be generated at random). Then indicate the pointers (high-light them) that need to be traversed in the process of searching for 22.

   33(4), 22(1), 15(3), 7(4), 19(2), 5(2), 40(1), 30(4), 9(1), 32(3).

6. (5 marks) Draw an AVL tree of height 3 with a minimum number of nodes.

7. (15 marks) Balanced trees. Consider the following AVL tree T. Perform the following operations. For each operation, use the **original** tree T. You must draw the tree after insertion each step and *indicate* what operation (such as single rotation, double rotation) you have performed.

(a) Insert 13 into T.

(b) Delete 1 from T (use the original T).

(c) Delete 12 from T (use the original T).

8. (10 marks) A singly-linked list can end either with a nil-pointer, or with a pointer going back to an earlier node in the list (not necessarily the first). In the latter case, we say the list is *cyclic*. Give an algorithm that determines whether a given list is cyclic or not. Try to find an algorithm with asymptotically optimal running time and using only a constant amount of space.

   Clearly describe your algorithm (not program) first. Then give its time complexity in terms of the number of nodes in the list (no proof needed). Correct but suboptimal algorithms receive partial marks.