Comprehensive Exam on Data Structures and Algorithms: Spring 2006

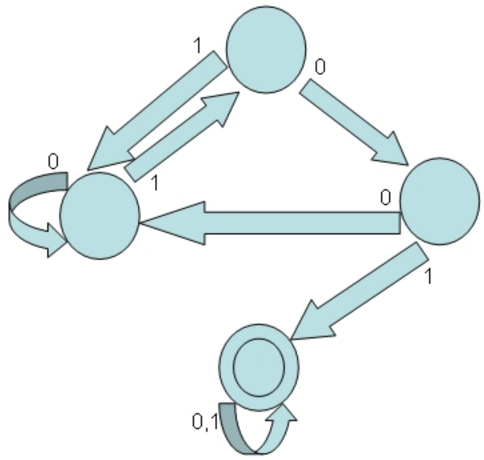
Instructions: Choose any four of the following six questions to answer. Clearly mark which question you want graded on the front of the exam by placing an X in the appropriate slot.

Problem	1	2	3	4	5	6
Graded?						

Note that you will be graded not just on the answers, but also on work. An answer alone will not be worth much if anything at all. No books, calculators, notes, talking, etc. Using any communications device at all (cell phones, walkie talkies, etc.) will be an automatic failing grade.

WARNING: If you do not mark the questions you want graded or mark them ambiguously (i.e. if you decide to mark 5 questions rather than 4, etc.), we will grade your lowest 4 questions.

1. Consider the following finite-state machine.



(a) Is the following string accepted or rejected by the above machine (note: the start state is the state on the far left)?

00110011

(b) Is the following string accepted or rejected by the above machine?

00101011

(c) Describe all bit strings that the above machine accepts.

2. Given a simple graph $G = (V, E)$ and a nonnegative integer K, the Vertex Cover decision problem is to decide if G has K edges $e_1, e_2, \ldots, e_K$ for which every vertex $v \in V$ is incident with at least one of the K edges. It is known that Vertex Cover is NP-Complete.

   Given a collection of m-bit binary strings $s_1, s_2, \ldots, s_n$, and a nonnegative integer K, the String Sum decision problem is to decide if there are K strings $s_{i_1}, s_{i_2}, \ldots, s_{i_K}$ from the collection such that

$$s_{i_1} + s_{i_2} + \cdots + s_{i_K} = \mathbf{1},$$

   where the addition is bitwise Boolean-algebra addition (e.g. $1001 + 1010 = 1011$), and $\mathbf{1}$ is the m-bit binary string consisting of all ones. Prove that String Sum is an NP-complete decision problem.

3. Consider the following algorithm for evaluating the polynomial

$$\sum_{i=0}^{n} a_i x^i$$

Algorithm: Let $ans = a_n$. For $j = n - 1$ downto 0, multiply ans by x, then add a_j to ans. This algorithm is also known as Horner's algorithm.

(a) Show the steps of Horner's Algorithm for evaluating the polynomial

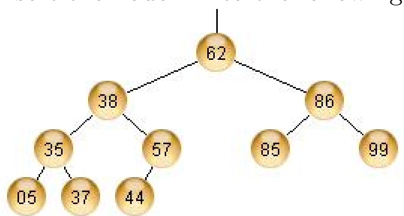
$$8x^5 - 17x^4 + 7x^3 - 5x^2 - 2x + 4$$

for the value $x = -1$.

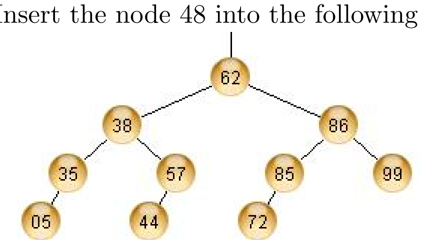
(b) Write pseudocode for a recursive version of Horner's algorithm.

4. Assume that the denominations of a certain country are 1 cent, 2 cent, and 4 cent pieces. Prove that the greedy algorithm always gives the smallest number of coins for change for any amount of money.

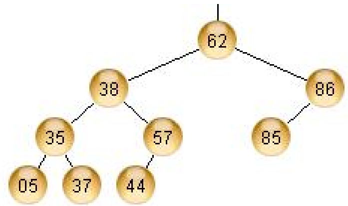
5. (a) Insert the node 2 into the following AVL tree.



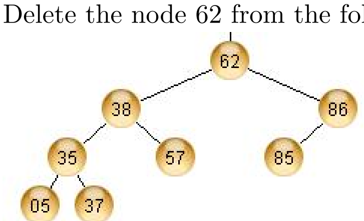
(b) Insert the node 48 into the following AVL tree.



(c) Delete the node 86 from the following AVL tree.



(d) Delete the node 62 from the following AVL tree.



6. (a) Show how to multiply two linear polynomials $ax + b$ and $cx + d$ using only three multiplications. (HINT: One of the multiplications is $(a + b)(c + d)$.)

(b) Assume that there exists a divide-and-conquer algorithm for multiplying two polynomials of degree n that runs in time $\Theta(n^{\log_2 3})$. Show that two n-bit integers can be multiplied in $O(n^{\log_2 3})$ time, where each step operates on at most a constant number of 1-bit values.