DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

PhD Qualifier Examination, Paper I

Total time: 3 Hours **March 08, 2011** **Maximum Marks: 120**

Answer ALL the parts

Part A: Discrete Mathematics

Answer ALL questions

A.1 Prove by induction that there are at most 2^h leaves in a binary tree of height h. **(6)**

A.2 For a given set A, consider the relation

$$R = \{(x, y) \mid x \in \mathcal{P}(A),\ y \in \mathcal{P}(A) \text{ and } x \subseteq y\},$$

where $\mathcal{P}(A)$ denotes the power set of the set A. Show that R is a partial order relation. **(6)**

A.3 (a) How many numbers between 1 and $10,000$ are not divisible by 6, nor by 9, nor by 14? **(6)**

(b) There are ten pairs of shoes in a closet. If eight shoes are chosen at random, what is the probability that no complete pair of shoes is chosen? **(7)**

A.4 (a) Let n lines be drawn on the plane so that no two lines are parallel and no three lines meet at a single point. Find a recurrence relation for *the number of infinite regions* created by the given lines, and solve the recurrence. **(8)**

(b) How many solutions are there to the equation

$$x_1 + x_2 + x_3 + x_4 = 17,$$

where x_1, x_2, x_3, x_4 are non-negative integers? **(7)**

Part B: Algorithms

Answer ALL questions

B.1 A sorted array A with n elements is cyclically right-shifted by k positions. For example, the sorted array $3, 9, 11, 12, 17$ cyclically right-shifted by 2 positions is the array $12, 17, 3, 9, 11$. You are given the right-shifted array A (and n), but not k. Design an algorithm to find out the maximum element of A in $\mathrm{O}(\log n)$ time. **(12)**

B.2 You are given an unsorted array $A = \{a_1, a_2, \ldots, a_n\}$ of n elements and a target sum T. Your task is to locate two elements a_i, a_j in A (with $i \neq j$) such that $a_i + a_j$ is as large as possible, but no larger than T. Consider the following $\mathrm{O}(n)$-time greedy algorithm to solve this problem.

> Initialize $i = 1$, $j = 2$, and $S = a_1 + a_2$.
> For $k = 3, 4, \ldots, n$, repeat the following two steps:
> If $a_i + a_k \leq T$ and $a_i + a_k > S$, then assign $j := k$, and $S := a_i + a_k$.
> If $a_j + a_k \leq T$ and $a_j + a_k > S$, then assign $i := k$, and $S := a_j + a_k$.
> If $S > T$, return ***failure***, else return (a_i, a_j).

Prove or disprove: The above greedy algorithm always outputs the pair (a_i, a_j) with the maximum possible sum $a_i + a_j \leq T$. **(10)**

B.3 You are given a directed acyclic graph $G = (V, E)$ and two vertices $u, v \in V$. Design an $O(|E|)$-time algorithm to compute the number of paths from u to v. **(12)**

B.4 (a) Let P_1, P_2 be computational problems. What is meant by the statement: "P_1 *is polynomial-time reducible to* P_2"? **(3)**

(b) Given an algorithm A to solve a computational problem P, when can it be said that A is a time-optimal algorithm for P? **(3)**

Part C: Formal Languages and Automata Theory

Answer ANY FOUR questions

C.1 (a) Find a string of *minimum* length in $\{a, b\}^*$ *not* in the language corresponding to the regular expression $(a^* + b^*)(a^* + b^*)(a^* + b^*)$. **(3)**

(b) Give a regular grammar for the language

$$L = \{w \mid w \in \{0, 1\}^* \text{ and } w \text{ does } \textit{not} \text{ contain the substring } 000\}.$$ **(7)**

C.2 Give a DFA for recognizing all strings over $\Sigma = \{0, 1\}$ with at most one pair of consecutive 0's *and* at most one pair of consecutive 1's. **(10)**

C.3 Show that the language $L = \{0^m 1^n 0^m \mid m, n \geq 0\}$ is not regular. **(10)**

C.4 (a) Give a CFG G over the alphabet $\Sigma = \{a, b\}$ such that

$$L(G) = \{x \in \Sigma^* \mid n_a(x) > n_b(x)\},$$

where $n_a(x)$ denotes the number of a's in x, and $n_b(x)$ denotes the number of b's in x. **(6)**

(b) Given that "if L_1, L_2 are CFLs, then $L_1 \cup L_2$ is a CFL", give a CFG for the language

$$\{x \in \{a, b\}^* \mid n_a(x) \neq n_b(x)\}.$$ **(4)**

C.5 (a) Describe a PDA (an automaton with one pushdown store) for accepting (both even- and odd-length) palindromes. Briefly explain its working by giving the steps of computation—it is NOT necessary to give the state transition diagram. **(7)**

(b) Define undecidable languages. Give an example of an undecidable language—there is NO need to justify the undecidability of the language in your example. **(3)**
