

Theory Qualifier Exam

Your number: _____

Time limit: 2.5 hours. Use only the notes supplied by the Department

FALL 2009, CS DEPARTMENT, IIT

For every question, please write your answer in a clean and concise way. Use additional pages, start a new page with each problem and write only one side of the paper.

If you are asked to write an algorithm for a question, you have to write the **pseudo-code** of your algorithm and also put explanations about your pseudo-code. Also show correctness and estimate the running time.

Use procedures if you want – marking clearly what the parameters are and what they do, and with what running time in terms of its parameters. Unless the procedures are from the textbook, write pseudocode for the procedures.

1. Suppose we are given a n -element sequence S such that each element in S represents a different vote in an election, where each vote is given as an integer representing the ID of the chosen candidate. Suppose you know the number $k < n$ of candidates running. Describe a $O(n \log k)$ algorithm for determining who wins the election.

2. Assume that each node v in a binary search tree has three pointers, to parent, left child, and right child.

Design an algorithm (write it as a function, called **successor**), which, given a node v , finds w , the node-successor of v in **preorder**.

Analyze the running time of s consecutive calls to **successor** (that is, w is given as the argument to the next call, and so on) in terms of s and h , the height of the tree. A tight (that is, impossible to improve by more than a constant factor – use Big-Oh) analysis is worth one third of the points.

3. The Set Coverage problem has as input a collection of m sets S_1, S_2, \dots, S_m and positive integers K and R , and asks if there exists K indices $1 \leq i_1 < i_2 < \dots < i_K \leq m$ such that $|\cup_{j=1}^K S_{i_j}| \geq R$. Prove that Set Coverage is NP-complete, using only the handout (precisely, the NP-Complete problems given there).

4. Suppose we are given a *timetable*, which consists of:

- A set \mathcal{A} of n airports, and for each airport $a \in \mathcal{A}$, a minimum connecting time $c(a)$
- A set \mathcal{F} of m flights, and the following, for each flight $f \in \mathcal{F}$:
 - Origin airport $a_1(f) \in \mathcal{A}$
 - Destination airport $a_2(f) \in \mathcal{A}$
 - Departure time $t_1(f)$
 - Arrival time $t_2(f)$.

Describe a $O(n^2 + m \log m)$ (or faster) algorithm for the Flight Scheduling problem. In this problem, we are given airports a and b and a time t , and we wish to compute a sequence of flights that allows one to arrive at the earliest possible time in b when departing from a at or after time t . Minimum connection times at intermediate airports must be observed.

Describe your data representation. Give pseudocode, analyze the running time and prove correctness.