

Comprehensive Exam – Algorithms and Data Structures  
Spring 2007

General directions: Answer 3 of the following 4 questions. Each problem is worth 10 points.

Before giving more detailed descriptions of algorithms, describe in a brief (1-3 sentence) paragraph the main ideas and techniques. Complete pseudo-code is not necessary as long as you clearly specify how the algorithm works. You may use without proof any well-known algorithms and lower bounds for standard problems, as long as you state precisely and correctly the known result.

Give at least an informal proof for all answers; for algorithms, this should include some convincing argument of correctness, and a time analysis. The time complexity of the algorithms you present will be given some weight in your score, as well as your correctness proofs and run-time analysis. For example, a correct  $O(n^2)$  algorithm might be worth only 8 or 9 points if there is an  $O(n \log n)$  algorithm. The exact weight of efficiency versus correctness is given after the specific problems, if appropriate.

**I. Soda supply and demand.** Chez Bob has  $n$  different kinds of soda and has a supply of  $S_i$  cans for each kind  $i$ . There are  $m$  customers, and each customer  $j$  wants to drink at most  $D_j$  cans, but will only drink from a fixed set  $C_j$  of kinds of soda. Given these constraints, Chez Bob wants to find an assignment giving each customer some of the cans, so as to maximize the total number of soda cans sold.

Give an efficient algorithm for finding an optimal assignment.

**II. Finding a dense subgraph.** You are given an undirected graph  $G = (V, E)$  in which each node has degree at most  $2k$ . You want to find a dense subgraph: a subset of nodes  $S \subset V$  such that each node  $u \in S$  has edges to least  $k$  other nodes in  $S$ .

Given  $G$  and  $k$ , show how to efficiently find a dense subgraph, if one exists.

**III. Completing a partial sort.** You are given an array of integers  $A[1 \dots n]$  that is partially sorted in ascending order, in the following sense: there are only a small number, say  $k$ , of locations at which  $A[i] < A[i - 1]$ . Give an efficient algorithm for completely sorting this array; you should express its running time in terms of  $n$  and  $k$ .

**IV. One-dimensional facility location.** A wholesale firm serves  $n$  different retail shops, all along a main highway, with shop  $i$  at mile  $d_i$  along the highway. (Distances are linear, so a shop at  $d_i$  and one at  $d_j$  are a distance  $|d_i - d_j|$  from each other.) The firm will designate  $t$  of these shops as distribution centers; and each distribution center can serve shops up to distance  $D$  away.

Give an efficient algorithm for finding the maximum number of shops that can be served (using just  $t$  distribution centers).