NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

SEMESTER I: 2012–2013

EXAMINATION FOR

CS3230 – DESIGN AND ANALYSIS OF ALGORITHMS

November 2012 – Time Allowed 2 Hours

INSTRUCTIONS TO CANDIDATES

1. This examination paper consists of FOUR (4) questions and comprises THREE (3) printed pages (including this page).
2. Answer **ALL** questions.
3. This is an **Open Book** examination.

Question 1. (10 marks)

We are given two sets of natural numbers (all greater than or equal to 1) $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_n\}$. All elements in A are distinct. We say that B covers A if for every element $a_i \in A$ there is an element $b_j \in B$ such that $\sqrt{b_j} \leq a_i \leq b_j^2$. We are supposed to determine if B covers A in time $O(n \log n)$.

(a) (4 marks) Write idea of an algorithm for this task.

(b) (2 marks) Write the pseudocode.

(c) (2 marks) Argue the correctness of your algorithm.

(d) (2 marks) Argue that the running time is as desired.

Question 2. (10 marks)

We are given an n node labeled tree T of depth d and degree at most 4. Each node v of T is labeled by a real number $l(v)$. A node v of T is a local-maximum if $l(v)$ is greater than or equal to $l(w)$ for all nodes w that are joined to v by an edge. The labels of the nodes are hidden. We have access to a function f such that for all pairs of nodes (v_1, v_2), $f(v_1, v_2) = 1$ if $l(v_1) \geq l(v_2)$ and 0 otherwise.

We are supposed to determine a local maximum using $O(d)$ invocations of f.

(a) (5 marks) Write idea of an algorithm for this task. (Hint: You may start by comparing the label of the root with its children.)

(b) (2 marks) Write the pseudocode.

(c) (3 marks) Argue that the number of invocations of f is $O(d)$.

Question 3. (10 marks)

We are given a set S of n pairs of real numbers $S = \{(a_1, b_1), \ldots, (a_n, b_n)\}$ such that $a_1 \leq a_2 \leq \ldots \leq a_n$. We say that a pair of indices (i, j) is compatible if $|a_j - a_i| \geq |b_j - b_i|$. We are required to select a set $T \subseteq \{1, 2, \ldots, n\}$, containing n, of as large size as possible such that for any $i, j \in T$ the pair (i, j) is compatible.

(a) (5 marks) Give idea of a dynamic-programming algorithm running in time $O(n^2)$ to find the size of optimal T. Justify the correctness of the recursion relation you obtain.

(b) (3 marks) Write the pseudocode.

(c) (2 marks) Argue that the running time is $O(n^2)$.

Question 4. (10 marks)

For NP-complete problems, sometimes we wonder if special cases of them are easy to solve. We know that Vertex-Cover is an NP-complete problem. We may wonder if the problem becomes easy to solve in graphs of small degree. It turns out the vertex cover is a hard problem even for graphs of small degree. Consider the problem Degree-utmost-4-vertex-cover defined as follows.

Given a graph $G = (V, E)$ with degree at most 4 and a number k, does this graph have a vertex cover of size at most k ?

Show that Degree-utmost-4-vertex-cover is NP-complete.

END of QUESTIONS