Homework 2

Fragments of Structural and Algorithmic Graph Theory
University of Helsinki, September 2015

The homework is due **Friday September 11, 2015**. Solutions should be handed in pdf by e-mail to `tomescu@cs.helsinki.fi`. If you are taking the course for 2 credits, submit solutions to problems 1 and 2. If you are taking the course for 3 credits, submit solutions to all three problems. Write clearly. The solution sheet should contain the following data: first name, last name, university, student number, number of credits (2 or 3). If you are a student of a university other than the University of Helsinki, write also the name and contact details of the person responsible for your grade registration. You are supposed to solve the homework on your own.

1. *(Graph traversal algorithms and their applications.)*

   (a) Analyze the time complexity of breadth-first search if the graph G is given by the adjacency matrix.

   (b) Describe an algorithm with time complexity $O(|V|)$ that determines whether a given graph contains a cycle. Justify the correctness of your algorithm.

   (d) Prove or disprove by giving a counterexample: If a directed graph D contains cycles, then the algorithm for topological sort described in class produces a vertex ordering that minimizes the number of "bad" edges that are inconsistent with the ordering produced.

2. *(Application of* 2*-SAT.)*

   Consider the following problem:

   PARTITIONED VERTEX COVER:

   Input: A graph $G = (V, E)$, a partition $V = V_1 \cup \ldots \cup V_k$ (that is, $V_1 \cup \ldots \cup V_k = V$, and $V_i \cap V_j = \emptyset$ if $i \neq j$).

   Question: Does G admit a vertex cover C such that $|C \cap V_i| \leq 1$ for all $i \in \{1, \ldots, k\}$?

   Show that the problem is polynomially solvable, by reducing it to an instance of 2-SATISFIABILITY.

****

3. *(Graph traversal algorithms and their applications, continued.)*

   (a) Similarly as we can perform an extended version of Depth-First Search on disconnected graphs (as described in class), we can also perform an extended version on DFS on digraphs in which not all vertices are reachable from s. (This way, we obtain a set of edge-disjoint depth-first trees.) Give an example of a vertex u in a directed graph D which ends up in a depth-first tree containing only u, even though u has both incoming and outgoing edges in D.

   (b) Describe an algorithm with time complexity $O(|V| + |A|)$ for the following problem:
   Given an acyclic digraph $D = (V, A)$ and two vertices $s, t \in V$, compute the number of all s-t (directed) paths in D.
   Base your algorithm on a topological sort. (You may assume that the graph is given with lists of in-neighbors ($N^-(v)$) and out-neighbors ($N^+(v)$) for all vertices $v \in V$.) Justify why the time complexity is linear.