Problem Set no. 2 — Minimum directed spanning trees

Given: May 5, 2009

Exercise 1.1 Exhibit a directed graph in which the lightest edge participates in a directed spanning tree rooted at some vertex r, but not in a *minimum* directed spanning tree rooted at r.

Exercise 1.2 Exhibit a directed graph in which all *minimum* directed spanning trees rooted at some vertex r contain the heaviest edge in the graph, even through there are directed spanning trees rooted at r that avoid that edge.

Exercise 1.3 Exhibit a directed graph $G = (V, E)$ with two weight functions $w_1, w_2 : E \to R$ on it such that for every two edges $e_1, e_2 \in E$ we have $w_1(e_1) \leq w_1(e_2)$ if and only if $w_2(e_1) \leq w_2(e_2)$ and yet the minimum directed spanning trees rooted at some vertex r with respect to w_1 and w_2 are different.

Exercise 1.4 Describe a *deterministic* linear time algorithm for finding a minimum directed spanning tree in an *acyclic* graph.

Exercise 1.5 Give simple linear time reductions between the problems of finding a minimum directed spanning tree rooted at a given vertex r, and the problem of finding a minimum directed spanning tree rooted at an arbitrary vertex of the graph.

Exercise 1.6 Let $G = (V, E)$ be a directed graph with a weight function $w : E \to R$ defined on its edges. A *branching* B of G is a set of edges $B \subseteq E$ such that the subgraph (V, B) is acyclic and the indegree of each vertex in it is at most 1. Obtain an efficient algorithm for finding a *maximum* branching of G, i.e., a branching B for which $\sum_{e \in B} w(e)$ is maximized. (Note that some of the edge weights may be negative.)