B505: Applied Algorithms

HW5 (Due: **Apr. 23, Monday, 5pm**)

http://darwin.informatics.indiana.edu/col/courses/B505-18

(This HW contains 20 bonus points if you answer all questions correctly.)

1. (10 pts) Show how to find the maximum spanning tree of a graph, which is the spanning tree of the largest total weight.

2. (20 pts) Consider an undirected graph G = (V, E) with nonnegative edge weights w_e. Suppose that you have computed a minimum spanning tree of G, and that you have also computed shortest paths to all nodes from a particular node s. Now suppose each edge weight is increased by 1; the new weights are $w'_e = w_e + 1$.

(a) Does the minimum spanning tree change? Give an example where it changes or prove it cannot change.
(b) Do the shortest paths change? Give an example where they change or prove they cannot change.

3. (15 pts) Consider the following algorithm.

   Input: a weighted graph G=(V, E) and w.
   Maybe-MST(G, w)
   Sort the edges into non-increasing order of edge weights w;
   T ← E
   for each edge e ∈ E, taken in non-increasing order by weight
       if T-{e} is a connected graph
           T ← T-{e}
   Output T

   Either prove that the output of the algorithm is a minimum-spanning tree or give a counterexample when the algorithm does not output a minimum-spanning tree.

4. (20 pts) The diameter of a tree T=(V, E) is defined as the largest length of all shortest path (i.e., the path with the smallest number of edges) between pairs of vertices. Devise an O(|V| + |E|) algorithm to compute a diameter for a given tree.
5. (20 pts) Dr. Fuzzy suggests the following algorithm for finding the shortest path from node s to node t in a directed graph with some negative edges: add a large constant to each edge weight so that all the weights become positive, then run Dijkstra's algorithm starting at node s, and return the shortest path found to node t. Is this a valid method? Either prove that it works correctly, or give a counterexample.

6. (20 pts) Decide whether you think each of the following statements are true or false. Justify your answers.
   a. Let G be an arbitrary flow network with a source s, a sink t and a positive

integer capacity c(e) on each edge. If f is a maximum s-t flow in G, then f saturates every edge out of s with flow, i.e., for all edges e out of s, we have f(e)=c(e).

b. Let G be an arbitrary flow network with a source s, a sink t and a positive integer capacity c(e) on each edge. Let (A, B) be a minimum s-t cut w.r.t. these capacities. Now suppose we add 1 to every capacity, then (A,B) is still a minimum s-t cut w.r.t. these new capacities, i.e., c'(e)=c(e)+1 for every edge e.

7. (15 pts) Consider the following scenario. Due to large-scale flooding in a region, parametics have identified a set of *n* injured people distributed across the region who need to be rushed to hospitals. There are *k* hospitals in the region and each of the n people needs to be brought to a hospital that is within a half-hour's driving time of their current location (so different people will have different options for hospitals, depending on where they are right now).

   At the same time, one does not want to overload any one of the hospitals by sending too many patients its way. The parametics are in touch by cell phone, and they want to collectively work out whether they can choose a hospital for each of the injured people in such a way that the load on the hospital is *balanced*: each hospital receives at most *n/k* people. Give a polynomial-time algorithm that takes the given information about the people's and the hospitals' location, and determines whether this is possible.