Algorithms
Computer Science 140 & Mathematics 168
Spring 2012
Homework 9a
Due Thursday, March 22

On this assignment, as usual, you may appeal to any results that we've proved in class without reproving them. If you wish to use any such result, just state the result that you're using.

1. [**30 Points**] **Hurts' On-board Navigation System Revisited!** Designing all new algorithms is sometimes necessary, but often times we can use one or more existing algorithms to solve our problem. The advantage of using existing algorithms is that we already have proofs of their correctness and analyses of their running times. In contrast, developing all new algorithms requires that you prove correctness and derive the running times from scratch. This problem, brought to you by Hurts Car Rental, is a good example of how using existing algorithms can be the secret to all happiness!

   Hurts Car Rental has designed a new generation of alternative fuel vehicles. The new vehicles use a special fuel comprising a finely minced mixture of the "Algorithms" textbook, chocolate fudge Pop Tarts, Spam, Spam, Spam, and Mountain Dew. Due to this rather unusual fuel requirement, there are only certain cities in the country where the vehicles can be refueled. Thus, to get from the start city to the destination city, the driver must plan a route that ensures that the car can be refueled along the way. (Note that this problem differs in several important ways from the first Hurts problem on Assignment 3b. First, we no longer use fuel packs - we can now fill up with fuel like a normal car. Second, the graph is no longer a straight line - it's a general graph. Third, we only care about minimizing the distance travelled, not about the cost of fuel.)

   The on-board computer on such a vehicle contains a weighted directed graph in which the vertices represent cities, the directed edges represent one-way roads, and the weights on the edges represent the (strictly positive) lengths of the roads. The graph, of course, can have cycles!

   The computer knows which select cities have filling stations. To use the navigation system, the user will enter the starting city, the destination city, and the range of the vehicle on a full tank. (You may assume that the starting city and the destination cities, being Hurts rental cities, both have filling stations - and that the car starts with a full tank of fuel.) The computer will respond with the *shortest* route from the starting city to the destination city that ensures that the vehicle doesn't run out of fuel or determines that no such route exists.

   You should assume that there is some constant C such that every vertex has in-degree and out-degree (number of entering edges and number of exiting edges, respectively) at most C.

(a) **Describe** and **analyze the running time** of an efficient algorithm for determining such a path. There are several ways to do this problem, but if you break this problem up into a couple intermediate steps you will probably find that the important problem of proving correctness (in the next part of the problem) is relatively easy.

(b) *Prove* that your algorithm finds an optimal solution. Be careful and rigorous. If you use existing algorithms, you can assume that they do their part of the job correctly!