Fall 2001 **CMSC 451: Project Part 1** Samir Khuller

Due in class: Nov 27.

The project will be in two parts. The first part involves the design of an algorithm. The second part involves implementing the algorithm. You will turn in a solution to each part separately. Part 2 will be due on Dec 6. If you are unable to do part 1, you will receive an outline of the algorithm on Nov 28. You can still do the second part. Each part is worth 50%.

You are planning a long road trip to Los Angeles. We wish to find the shortest route to LA, but are not willing to spend more than C \$'s in tolls.

We can model this problem as follows: given a directed graph, where each edge has a *length* and a *cost* we wish to find the shortest length path from s to t (given an arbitrary pair of vertices) such that the total *cost* of the path does not exceed C. Moreover, we will assume that the cost of an edge is an integer. You have to design an algorithm with running time $O(f(n,m)C)$ where $f(n,m)$ is a polynomial function of the number of vertices n and the number of edges m in the network.

The input to the problem is: a directed graph G represented as an adjacency list (we will discuss detailed input format in Part 2), a pair of vertices s,t, and an integer C. You have to output the shortest length path from s to t which has cost at most C.