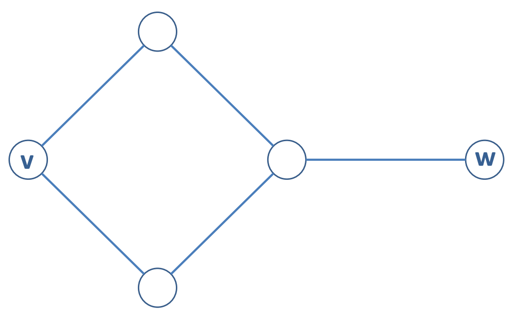
CS 577: Introduction to Algorithms	Homework 3
Out: 09/29/10	**Due: 10/06/10**

Ground Rules

- **Grading.** You will be graded on the correctness as well as clarity of your solutions. You are required to prove any claims that you make. In particular, when you are asked to design an algorithm, you must argue its correctness as well as running time. You may use without proof any theorems proven in class or in the textbook, as long as you state and cite them properly.
- **Collaboration.** You may work on and submit solutions in pairs.
- **Lateness.** Homework is due promptly at the start of class. Late homework will receive zero credit.
- **Extra credit questions.** There are no extra credit questions on this homework.
- Start working on your homework early. Plan your work in such a way that you have the opportunity to put some problems on the back burner for a while and revisit them later. Good luck!

Problems

1. **(5 points.)** Often there are multiple shortest paths between two nodes of a graph. Give a linear-time algorithm for the following task: given an undirected graph $G = (V, E)$ with unit edge lengths and nodes v and w, output the number of distinct shortest paths from v to w. For example, for the graph below, on input v and w your algorithm should output 2. (See problem 3.10 in the book for the "story" behind this question.)



2. **(2+4 points.)** The goal of this question is to design an algorithm for finding a shortest cycle in a given undirected graph with unit edge lengths. Here the length of a cycle is the number of edges in the cycle.

   (a) Consider the following algorithm: Run DFS on the graph starting from an arbitrary root s, and for each node, record its *level* in the DFS tree: its distance along the tree from the root s. Whenever we encounter a *back edge* (i.e. a non-tree edge) (u, v) during the algorithm, this edge forms a cycle with tree edges from u to v; the cycle has length $1 + |\text{level}(u) - \text{level}(v)|$. So as we run the algorithm, we keep track of the back edge with the shortest such cycle length seen so far and return the shortest one in the end.

   Show that this algorithm is incorrect. That is, provide an example where this algorithm returns the wrong answer as well as a brief explanation of it.

   (b) Give an algorithm for finding a shortest cycle in an undirected graph with unit edge lengths. Your algorithm should run in time $O(m^2)$ where $m = |E|$. Give a proof of correctness for the algorithm. You may assume that the graph is connected.

3. **(4 points.)** Problem 4.5 in the textbook (Pg. 190): reproduced below.

   Let's consider a long, quiet country road with houses scattered very sparsely along it. (We can picture the road as a long line segment, with an eastern endpoint and a western endpoint.) Further, let's suppose that despite the bucolic setting, the residents of all these houses are avid cell phone users. You want to place cell phone base stations at certain points along the road, so that every house is within four miles of one of the base stations.

   Give an efficient algorithm that achieves this goal, using as few base stations as possible.