Homework 11

Due Wednesday, May 4th at 11:59 pm ET on Gradescope

Solution guidelines. For problems that require you to provide an algorithm, you must give the following:

1. a precise description of the algorithm in English and pseudocode (*),
2. a proof of correctness,
3. an analysis of the asymptotic running time and space.

You may use algorithms from class as subroutines. You may also use any facts that we proved in class, e.g. correctness of subroutines, running time of subroutines. You should be as clear and concise as possible in your write-up of solutions.

(*) It is fine if the English description concentrates on the high level ideas and doesn't include all the details. But the reader should not have to figure out your solution solely based on the pseudocode. You can also add comments to your pseudocode, in fact that is best practice. FYI we will share a document on good pseudocode style with you and it will also be discussed in labs and lecture.

Problem 1. *Unique Min Cut (10 points)*

1. Give an example of a flow graph in which the minimum cut is not unique.
2. We usually find a minimum cut at the end of the Ford-Fulkerson algorithm by taking one side to be all nodes reachable from s in the residual graph. Prove that we can also find a min cut by taking one side to be all nodes with a path to t in the residual graph.
3. Give a polynomial-time algorithm to decide whether a flow graph has a *unique* minimum cut.

Problem 2. *Top Secret (10 points)*

You are given a directed graph $G = (V, E)$ (picture a network of roads). Employees of a top-secret company reside at a certain collection of nodes $A \subset V$ (one employee per node), and have to get to work at a certain other collection of nodes $B \subset V$ (assume that A and B are disjoint). Any employee of the company can go to work at any site in B, but company policy states that employees have to avoid being seen in public together on their way to work. A set of "secret" routes is defined as a set of paths in G so that (i) each node in A is the start of one path, (ii) the last node on each path lies in B, and (iii) the paths do not share any edges.

1. Given G, A, and B, show how to decide in polynomial time whether such a set of secret routes exists.
2. Suppose we have exactly the same problem as in (a), but we want to enforce an even stronger version of condition (iii). Thus we change (iii) to say "the paths do not share any *nodes*."
   - (a) With this new condition, show how to decide in polynomial time whether such a set of secret routes exists.
   - (b) Provide an example with the same G, A, and B, in which the answer is yes to the question in 2.1 but no to the question in 2.2 (a).