CS 6150: Graduate Algorithms, Fall 2013
Homework 3
Due 5:00 pm Sunday, Nov. 10, 2013

Instructions

- ***(New!)*** Unless explicitly stated, proofs of correctness are required for any algorithm given as well as an analysis of the worst case complexity.
- ***(New!)*** There are points for making your algorithm as efficient as possible, unless explicitly stated to give a naive algorithm.
- Each student must **electronically** submit individual solutions for these homework problems. We strongly suggest typesetting your solutions using software such as LaTeX or Microsoft Word.
- Please carefully read the course policies linked from the course web site. If you have any questions, please ask during lecture or office hours, or post your question to the course newsgroup. In particular:
  - You may use any source at your disposal – paper, electronic, or human – but you must write your solutions in your own words, and you must cite every source that you use. In particular, each solution should include a list of everyone you worked with to solve that problem.
  - Unless explicitly stated otherwise, every homework problem requires a proof.
  - Answering "I don't know" to any homework or exam problem (except for extra credit problems) is worth 25% partial credit.
  - Algorithms or proofs containing phrases like "and so on" or "repeat this process for all n" instead of an explicit loop, recursion, or induction, will receive 0 points.
- If the question asks for a simple answer, you should still **show your work**. If you don't show your work there is no chance for partial credit.

Problems

---

1. (Graphs) In the town of Pongville, all residents play ping pong. Strangely enough, in a matchup of any given pair of players, the same player always wins. For example, if Josh plays Sarah, either Josh wins every game or Sarah wins every game. Even stranger, the overall graph can have cycles. That is, it may be that Josh always beats Sarah who always beats Frank who always beats Josh.
   (a) Prove that any finite set of residents of Pongville can be arranged in a row (called a "winning order") from left to right so that every player always beats the player immediately to its left. If there are no cycles, the resident that always loses will be on the left and the player that always wins will be on the right.
   (b) Suppose you are given a directed graph representing the win/loss relationships among a set of n players. The graph contains one vertex per player, and it contains an edge $i \to j$ if and only if player i beats player j. Describe and analyze an algorithm to compute a winning order for the players, as guaranteed by part (a).

2. (Max flow) Consider the following modification to the Ford-Fulkerson algorithm. Instead of maintaining a residual graph, just reduce the capacity of edges along the augmenting path! In particular, whenever we saturate an edge, just remove it from the graph.

```
Data: G, c, s, t
for every edge e in G do
|  f(e) ← 0
end
while there is a path from s to t do
|  π ← path(s, t)
|  F ← capacity of bottleneck edge in π
|  for every edge e in π do
|  |  f(e) ← f(e) + F
|  |  if c(e) = F then
|  |  |  remove e from G
|  |  else
|  |  |  c(e) ← c(e) − F
|  |  end
|  end
end
return f
```

(a) Show that this algorithm does not always compute a maximum flow.

(b) Prove that for any flow network, if an oracle tells you precisely which path π to use at each iteration, then the above algorithm does compute a maximum flow. (Sadly, such an oracle does not exist.)

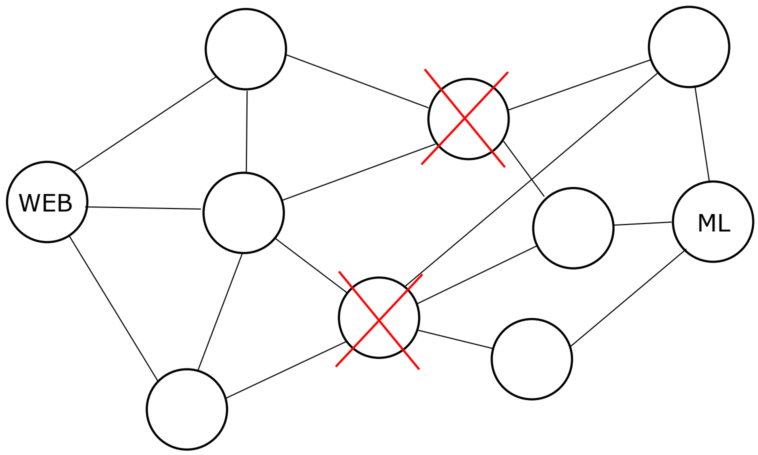
3. (Max flow) Unbeknownst to most students, all buildings on the University of Utah campus are connected by a secret network of underground tunnels that allow TAs to escape students that are angry about their homework grades.

Terrorists from BYU have infiltrated the WEB building and are spraying blue graffiti over all the walls. Unfortunately, these terrorists know about the underground tunnels and plan to use them to evenutally cover all of the buildings in blue. Thankfully a system is in place to block tunnel travel at buildings. That is, if the Huntsman Center is shut down, then no travel between tunnels connected at the Huntsman Center is possible.

Alan Administrator is a lazy fellow who lives for books and little else and sadly, he is in charge of containing the terrorists. Really all Alan cares about is the Marriott Library (ML), and so he plans to shut down as few buildings as possible to protect the ML. He can't shut down WEB because it is already infiltrated and won't shut down the ML because he plans to spend the afternoon there curled up with the latest Nora Roberts novel.

Describe and analyze an algorithm to find the minimum number of buildings that must be closed to block all tunnel travel from WEB to ML. The tunnel network is represented by an undirected graph, with a vertex for each building and an edge for each tunnel between two buildings.

For example, given the following input graph, your algorithm should return the number 2.



University of Utah tunnel network

4. (Max flow) The Center for Cultural Understanding at U of U has decided to convene a committee to determine how culturally sensitive the university is. This committee is to have exactly one student representative from every nation in the world. To broaden their choices, a student who has spent more than one year in a country is eligible to represent that country. For example, Henrietta Moussalem was born in Egypt, spent her primary school years in Tajikistan and attended high school in Bolivia, so she is eligible for all three countries.

   One additional requirement is that the committee must contain the same number of freshmen, sophomores, juniors and seniors. Fortunately the number of countries in the world is a multiple of 4 (well, we'll pretend for now that this is true), and there is at least one U of U student affiliated with every country.

   Describe an efficient algorithm to select the membership of the Cultural Sensitivity Committee. Your input is a list of all U of U students, their year (freshman, sophomore, junior or senior), and their country affiliation(s). There are n students and $4k$ countries. Discuss what will happen if no solution exists.

5. (Minimum spanning trees) Given an undirected weighted graph G, let $\overline{st}_i$ be the ith path from node s to node t among all paths from s to t (different paths need not be edge disjoint), and let $\overline{st}_i^j$ represent the weight of the jth edge in the path. Describe, prove correctness, and analyze an algorithm to find

$$\min_i \max_j \overline{st}_i^j$$

   between *every* pair of vertices in G. Assume no two edges have the same weight.