CS 473: Algorithms, Fall 2008
HW 8 (due Tuesday, November 4, 11am)

This homework contains four problems. **Read the instruction for submitting homework on the course webpage**. In particular, *make sure* that you write the solutions for the problems on separate sheets of paper and then staple them together. Write your name and netid on each sheet.

Collaboration Policy: For this homework you are allowed to work in groups of up to 3 students each. Starting this week, a third of the on-campus students will be presenting their homework orally. Please see the newgroup for instructions on which groups will be presenting orally and the instructions for signing up for a slot. The other groups will submit a written homework.

1. (25 pts) Problem 7.11 from the textbook.

2. (30 pts) Given a flow network G with integer capacities you have computed a maximum flow f between s and t. However you have made a mistake in noting the capacity of an edge e.

   - (10 pts) Suppose you *under* estimated the capacity of e by $k > 0$ units. Show that you can compute the correct maximum flow in $O(km)$ time using the current flow f.
   - (20 pts) Do the same as above if you *over* estimated the capacity of e by $k > 0$ units. *Hint:* First assume that f is acyclic. How do you reduce flow on e?

3. (25 pts) Let $G = (V, E)$ be a flow network with non-negative integer capacities; $c(e)$ is the capacity of $e \in E$. Given G and the source s and sink t, we want to know whether an edge e is *critical* for the flow between s and t. Formally, e is critical if *every* s-t minimum cut contains e.

   - (18 pts) Describe an algorithm that given G and e decides if e is critical for s, t. *Hint:* Suppose there is a minimum cut that does not contain e. How can you alter the capacity of e to find it?
   - (7 pts) Assuming you have an algorithm for the above problem, show how use it to check whether a given graph G has a *unique* minimum s-t cut. A minimum-cut (A, B) is unique if there is no other cut (A', B') of the same capacity and $A \neq A'$.

4. (20 pts) A graph is regular if every vertex has the same number of edges incident to it. Prove that every regular bipartite graph has a perfect matching.