CS 577: Introduction to Algorithms	Homework 6
Out: 04/24/12	**Due: 05/01/12**

Ground Rules

Same as for homework 1.

Problems

1. **(4 points.)** For this question, you are given a directed network $G = (V, E)$ with capacities c_e on edges and a source-sink pair (s, t). You are also given the max s-t flow in the graph, f^* specifying the amount of flow on every edge in the graph. An edge in the network is called a *critical* edge if reducing its capacity by one unit decreases the s-t max flow in the network.

   Develop an algorithm for determining all the critical edges in the given graph. Your algorithm is allowed to use the max flow f^* and should run in time $O(mn)$.

   (Hint: try your algorithm on the example in HW5.)

2. **(4 points.)** Problem 7.12 in the textbook (Pg. 420).

3. **(7 points.)** Problem 7.27 in the textbook (Pg. 431).

   Hint: As you may have guessed, you should reduce this problem to a network flow problem. Prove that a fair schedule exists if and only if there exists an integral max flow in your network of a certain size. Proceed to show (for part (a)) that the max flow has the right size. Note that in order to prove this, you only need to display a fractional flow of that size, and not necessarily an integral one (Ford-Fulkerson guarantees the existence of an integral flow of the same size).

4. **(Extra credit.)** Recall that a *vertex cover* of a graph $G = (V, E)$ is a subset of V, V', such that *every* edge in E is incident on some vertex in V', that is, for each edge $(u, v) \in E$, one or both of u and v are in S. Show that the problem of finding the minimum vertex cover in a *bipartite* graph reduces to maximum flow. (Hint: Can you relate this problem to the minimum cut in an appropriate network?)