Algorithms
Computer Science 140 & Mathematics 168
Spring 2009
Homework 9b
Due Tuesday, March 31

This problem set has one problem on flows and cuts and the remaining problems are on NP-completeness. When proving that a problem is NP-complete, keep in mind that there are several things that you need to do:

1. Prove that the problem is in the class NP.
2. Give a reduction from some known NP-complete problem to your problem.
3. Explain why your reduction takes polynomial time.
4. Prove that the reduction is correct (that is, if you are reducing from problem X to problem Y, that the instance of problem X is a "yes" instance if and only if the constructed instance of Y is also a "yes" instance). Notice the "if and only if" here. That requires two separate parts in your proof.

1. **[20 Points] Space Shuttle Profit Optimization!**

   After successful careers at Millisoft, Hurts Car Rental, the brokerage firm of Weil, Proffet, and Howe, and Giigle, you've been hired by NASA to help optimize their Space Shuttle program.

   On a given mission, NASA will consider a set of experiments that industrial sponsors would like to conduct (for example, "Does zero-gravity compromise the integrity of Spam?"). Let $E = \{E_1, E_2, \ldots, E_m\}$ denote the set of experiments under consideration. Let p_j denote the amount of money the sponsor will pay to conduct experiment E_j. The experiments use a set $I = \{I_1, I_2, \ldots, I_n\}$ of instruments to conduct the experiments. For each experiment E_j, let R_j be the subset of I that contains all of the instruments needed to conduct experiment E_j. Notice that this allows a single instrument to be used in multiple experiments. The instruments are potentially large and heavy and the cost of taking instrument I_k is c_k dollars. Your job is to determine which experiments should be performed in order to maximize the net revenue, which is the total income from the performed experiments minus the total cost of all instruments carried. Amazingly, this problem can be solved using **network flow**!

We'll construct a network flow problem as follows: The network contains a source vertex, s, vertices $I_1, I_2, \ldots, I_n$, vertices $E_1, E_2, \ldots, E_m$, and a sink vertex t. For each instrument I_k there is a directed edge from s to vertex I_k with capacity c_k (the cost of taking this instrument). For each experiment E_j there is an edge from vertex E_j to t with capacity p_j (the payment for this experiment). Finally, if instrument I_k is in set R_j (meaning that instrument I_k is needed for experiment E_j) then there is a directed edge from vertex I_k to vertex E_j with **infinite** capacity.

(a) First, try this. Consider a situation in which there are three experiments E_1, E_2, E_3 which will bring in 10, 6, and 6 dollars, respectively. Also there are four instruments I_1, I_2, I_3, and I_4 which cost 3, 2, 5, and 7 dollars, respectively to take on the shuttle. Experiment E_1 requires instruments I_1 and I_2, experiment E_2 requires instruments I_1 and I_3, and experiment E_3 requires instruments I_3 and I_4. Using brute-force, enumerate all seven possible combinations of experiments that could be taken and determine which combination is the most profitable. What is this combination and what is the net revenue?

(b) For the example problem above, construct the corresponding network flow problem. Find the maximum flow in the network (show your residual graphs at each step) and then find the corresponding cut whose capacity is equal to this flow.

(c) Let's let S denote the vertices that are on the same side of the above cut as vertex s and let T denote the vertices that are on the same side of the cut as t. What do you notice about the instruments and experiments that are in the set T?

(d) Now, let τ be the sum of the payments that NASA would receive for taking all of the possible experiments. In this case, $\tau = 10 + 6 + 6 = 22$. From τ, subtract the capacity of the cut you found above. Surprise! What is this number and how does it appear to relate to this problem?

(e) Finally, we're ready to generalize all of this into an efficient algorithm for solving the profit maximization problem in general. Assume that we've set up a network flow problem corresponding to a given set of experiments and instruments. Show that for any cut with finite total capacity, if an experiment E_j is in T (the side of the cut containing vertex t), then all of the instruments used in this experiment must **also** be in T.

(f) Now argue that the maximum net revenue that can be achieved is simply the total sum τ of the payments that would be received for taking all of the experiments minus the capacity of the minimum cut.

(g) Now just summarize all of this by describing the algorithm, step-by-step, for finding the maximum net revenue, given a set of experiments, instruments, and the corresponding payments and costs. Give a careful derivation of the worst-case running time of the algorithm assuming there are m experiments and n instruments.

2. [**5 Points**] **Professor Lai and NP!** Professor I. Lai from P.I.T. claims that if P_1 and P_2 are any two problems in NP such that $P_1 \leq_p P_2$, then $P_2 \leq_p P_1$. Prove that this would imply that P = NP.

3. [**15 Points**] **ILP Revisited!**

   In class we proved that INTEGER LINEAR PROGRAMMING (ILP) is NP-complete. Our reduction was from 3SAT. In this problem you will prove that ILP is NP-complete using a reduction from VERTEX COVER. You do not need to show that ILP is in NP (we already argued that in class).

4. [**15 Points**] **Hitting Set!** First, the gratuitous story: The Computer Science Department at The Western Institute of Technology (T.W.I.T.) has 5 faculty members, Dr. Theo Rize, Dr. I. Dutu, Dr. Sam Heer, Dr. Mia Thue, and Dr. Juan More. Each professor serves on a number of committees. The Polynomial Time Algorithms Committee consists of Professors Rize, Dutu, and More. The NP-Completeness Committee consists of Professors Dutu, Heer, and Thue. Finally, the Other Stuff Committee consists of Professors Thue and More.

   The Dean would like to convene a meeting in which the number of computer scientists invited is minimized (she can't stand their bad

CS jokes!) but each committee is represented by at least one of its members. For example, inviting Professors Thue and More would be a solution in this case.

The Dean has formulated this problem as the following general problem (known to computer scientists as the HITTING SET Problem): We are given a set S (the set of professors) and a collection, C, of subsets of S (the set of committees).

A Hitting Set for C is a subset $S' \subseteq S$ such that S' contains at least one element from each subset in C. The HITTING SET optimization problem is to find the smallest Hitting Set.

(a) Formulate the HITTING SET decision problem corresponding to the optimization problem described above.

(b) Prove that the HITTING SET decision problem is NP-complete.

5. **[20 Points] Dominating Set.** The DOMINATING SET problem is stated as follows: Given a graph $G = (V, E)$ and a positive integer ℓ, does there exist a subset S of ℓ vertices in V such that every vertex in V is either in S or is connected by an edge to some vertex in S.

   Take a close look at this problem to make sure that you understand how this problem differs from the seemingly similar VERTEX COVER problem. Prove that DOMINATING SET is NP-complete using a reduction from 3SAT.

6. **[20 Point OPTIONAL Bonus Problem!] Multicommodity Network Flow!**

   We know how to solve the network flow problem in polynomial time. A generalization of network flow is called *multicommodity network flow* and it goes like this: We are given a flow network except now there are multiple source/sink pairs in the network. That is, there are some number k of sources $s_1, \ldots, s_k$ and the same number of sinks $t_1, \ldots, t_k$. Each source s_i pushes a different kind of "fluid" or "commodity" to its corresponding sink t_i. However, the edges of the network are shared by these flows and each edge still has a single integer capacity. For example, an edge with capacity 5 might accommodate 2 units of flow from s_1 to t_1 and 3 units of flow from s_2 to t_2. Now we are interested in finding a set of flows, one for each source/destination pair. Of course, we are still subject to capacity constraints and conservation of flow (i.e. for any given commodity, the amount of flow for that commodity entering a given vertex is equal to the amount of flow for that commodity leaving that vertex).

It turns out that maximizing the total amount of flow in a multicommodity network can still be solved in polynomial time by using polynomial time linear programming algorithms! However, the optimal solution found by the linear program will, in general, not have integer flows.

If we now simply restrict the flows for each commodity to have integer values, the problem becomes NP-complete! Amazing, but true! In particular, the decision problem that we will consider here is this: Given is a flow network with k commodities and k source/sink pairs, (s_i, t_i) (one for each commodity), and a demand d_i for each sink t_i. The question is: "Does there exist a valid set of flows, one for each commodity, such that each sink receives d_i of commodity i?"

Prove that this problem is NP-complete. A reduction from 3SAT is particularly nice! (It turns out that this problem remains NP-complete even if there are just two source/sink pairs and thus two commodities, and each edge has capacity 1. You don't need to prove this stricter result, but it's kind of surprising that NP-completeness is maintained even under these very tight constraints!)

If you prefer, you may come and talk to me in person to show me your proof on my whiteboard. This will save you time in writing up your solution and will still allow you to claim your bonus points!