Problem Set 3— Due 5/27/2010, 3PM

(20) Problem 1. We consider an extension of problem 3 on the midterm where we wanted to find the cheapest r edge disjoint paths to r of a set D of k destinations $d_1, \ldots, d_k$.

Now suppose that we have a profit p_i associated with the *ith* destination. We want to select a subset D^* of D and a set P of edge disjoint paths to each element of D^* that is of maximum value. The *value* of a solution D^* with associated paths P is the sum of the profit of the destinations in D^* minus the total cost of the paths in P (in essence the profit for the destinations selected, minus the cost of getting to them).

.

Give an efficient algorithm to find the optimal sets D^* and associated paths P. Justify its correctness and give its run time. NOTE: you are no longer looking for a solution with r destinations. The optimal could have zero to k destinations.

(25) Problem 2. Entrepreneur's problem. An entrepreneur faces the following problem. In each of T periods, he can buy, sell, or hold for later sale some commodity, subject to the following constraints. In each period i he can buy at most α_i units of the commodity, can holdover at most β_i units of the commodity for the next period, and must sell at least γ_i units. Assuming that p_i, w_i, and s_i denote the purchase cost, inventory carrying cost, and selling price per unit in period i, what buy-sell policy should the entrepreneur adopt to maximize total profit in the T periods? Formulate this problem as a minimum cost flow problem for $T = 4$. (Assume start with no commodities so no requirement to sell any in the first period).

(25) Problem 3. A bus company has n morning runs and n afternoon runs with m_i, a_i the duration of the *ith* morning/afternoon run in hours. The goal is to assign drivers to each run (each driver does one morning run and one afternoon run). If a driver is assigned to runs that take more than D hours, then the company has to pay the driver an overtime bonus b for each hour worked above D (or if a part of an hour, then b times that fraction.) The company wants to assign drivers to minimize the total overtime.

a) Formulate this as a (weighted) bipartite matching problem.

b) Prove that if we sort the morning runs in nondecreasing order of length and the afternoon ones in nonincreasing order then assigning the *ith* driver to the *ith* morning and afternoon run is an optimal assignment.

(30) Problem 4. Consider the scheduling problem where we have n tasks each with a deadline d_i and a processing time p_i. All tasks are available at time zero (so identical release times). We want to schedule all jobs on m identical processors with no preemptions.

a) When $m = 1$ prove that we can always schedule the jobs in nondecreasing order of d_i value when the problem is feasible.

b) Prove that the m processor version of this problem is strongly NP-complete (that is, the input is m, n and the list of the d_i, p_i values and we want to determine if there is a feasible schedule).

c) Give an efficient solution to the problem of part b) if we allow preemptions.

(25) Problem 5. Problem 4.2 in the text