Spring 2010 **CMSC 858Y: Homework 3** Samir Khuller

Due in class: No due date - complete by April 4.

(1) Design a polynomial time algorithm to find a minimum weight vertex cover in a bipartite graph by directly reducing it to network flow. Nodes have weights, and we wish to find a minimum weight subset of nodes that cover all edges.

(2) Suppose a company has the following vehicle requirements. Jan: 230 cars, Feb: 210 cars, Mar: 440 cars, Apr: 390 cars, May: 425 cars, Jun: 450 cars. They have the following lease options. A 3 month lease for 1700$, a 4 month lease for 2200$, a 5 month lease for 2600$.

Show how to formulate this problem as a linear (integer) programming problem, and then argue that this can be reduced to a min cost flow computation.

(3) We have a periodic scheduling problem. There are $n$ tasks. Each task takes one unit of time to perform. The requirement is that task $i$ should be scheduled once in each time period $p_i$.

For example if task $i$, has period $p_i$. We must schedule it in each time-window of the form $[p_i \cdot j + 1, \ldots, p_i \cdot (j+1)]$ for $j = 0$.... Let $L = lcm(p_1, p_2, \ldots, p_n)$. Moreover suppose that $\sum_i \frac{1}{p_i} \leq 1$. Show how to find a periodic schedule for the first $L$ time slots. If the above condition is not satisfied, there is no periodic schedule. Why?

(4) Given an undirected unweighted graph $G$, we wish to compute a subgraph in which every vertex has degree at most 2. Moreover, we wish to maximize the number of edges in the subgraph. How can we reduce this to matching?