Due in class: Feb 28.

Problems are to be done by yourself (unless specified) with no help from another person. If you use any sources other than the textbooks for the class, you are to cite them. In any case, the writeup should be your own and not simply taken from another source.

(1) You are given an algorithm A that takes as input a graph $G$ with weights on the edges and computes a maximum weight matching $M$ (not necessarily perfect). Your goal is to compute a max weight perfect matching $M^*$. Show how you can do this by using algorithm A.

(2) Given an undirected unweighted graph $G$, we wish to compute a subgraph where every vertex has degree at most 2. Moreover, we wish to maximize the number of edges in the subgraph. (For example, if the degree constraint is 1, this is just the max matching problem.)

(3) Find a min weight edge cover of a simple undirected graph $G = (V, E)$. An edge cover is a collection of edges that cover all the vertices of the graph. In other words, it is a minimum weight subgraph in which each vertex has degree at least 1.

(4) You are given $n$ unit length jobs to schedule on a machine. Job $j$ has a deadline $d_j$ and weight $w_j$. If the deadline is met, then there is no penalty. If the job is scheduled after the deadline, we pay a penalty of $w_j(C_j - d_j)$, where $C_j$ is the completion time of Job $j$. Show that the problem of minimizing the total penalty can be solved in polynomial time.