Due: Wednesday, March 30, beginning of lecture

NOTE: Each problem set only counts 5% of your mark, but it is important to do your own work (but see below). These assignments will be followed by term tests, each worth 15% of your final grade. You may consult with others concerning the general approach for solving problems on assignments, but you must write up all solutions entirely on your own. You may chose to work in pairs and then submit one assignment but both partners should work on all questions. Anything else is *plagiarism*, and is subject to the University's Code of Behavior. You will receive 1/5 points for any question/subquestion for which you say "I do not know how to answer this question". You will receive .5/5 points if you just leave the question blank.

1. Show how to formulate the following problems as $\{0,1\}$ integer programming (IP) problems.

   (a) Given $\{(v_1, w_1), \ldots, (v_n, w_n)\}$ and a knapsack weight bound W, compute the maximum value $\sum_{i \in S} v_i$ for a feasible subset $S \subseteq \{1, \ldots, n\}$ where now S is feasible if $\sum_{i \in S} w_i \leq W$ and furthermore for any $i > 1$, if $i \in S$ then $i - 1 \notin S$.

   (b) Consider a variant of the set cover problem where we no longer have to cover every element in the universe $U = \{u_1, \ldots, u_n\}$ but an uncovered element u_j incurs a non negative penalty p_j. The problem then is as follows: Given a universe $U = \{u_1, \ldots, u_n\}$ with p_j = the penalty if element u_j is not covered, and given a collection $\mathcal{C}$ of subsets $S_i \subseteq U$ for $1 \leq i \leq m$ with w_i = cost of using S_i in a subcollection $\mathcal{C}'$, minimize the sum $A + B$ where A = sum of the costs of the subsets in $\mathcal{C}'$ and B = sum of the penalties of elements not in $\cup\{S_i | S_i \in \mathcal{C}'\}$.

2. Consider the $P1|r_j, prec|\sum w_i C_i$ scheduling problem; that is, jobs have a release times and there is a precedence relation $\leq_{prec}$ amongst jobs and the objective function is to minimize the weighted sum of completon times.

   (a) Extend the IP/LP formulation give for $P1|prec|\sum w_i C_i$ so as to formulate this problem as an IP/LP relaxation.

   (b) Extend the analsyis given for $P1|prec|\sum w_i C_i$ so as to obtain a 3-approximation algorithm.

3. Consider the following 3-colouring optimization problem: Given a graph $G = (V, E)$, a vertex 3-colouring $c : V \rightarrow \{1, 2, 3\}$ satisfies an edge $e = (u, v) \in E$ if $c(u) \neq c(v)$. We want to find a 3-colouring so as to maximize the number of edges satisfied.

   (a) Show how to use "the random method" to obtain a 3-colouring which in expectation satisfies $\frac{2}{3}|E|$ edges and hence is a $\frac{2}{3}$-approximation.

   (b) Show how to derandomize this algorithm.

4. Describe a randomized algorithm which given any polynomial $q \in Z_p[x]$ of degree 3 will output the complete factorization of q with expected running time $(\log p)^{O(1)}$. Justify the correctness and expected running time of your algorithm.