CS 573: Graduate Algorithms, Fall 2010
Homework 5

Practice only — Do not submit solutions

---

1. (a) Describe how to transform any linear program written in general form into an equivalent linear program written in ***slack*** form.

$$\begin{array}{|lll|}\hline \text{maximize} & \displaystyle\sum_{j=1}^{d} c_j x_j & \\ \text{subject to} & \displaystyle\sum_{j=1}^{d} a_{ij} x_j \le b_i & \text{for each } i = 1 \,..\, p \\ & \displaystyle\sum_{j=1}^{d} a_{ij} x_j = b_i & \text{for each } i = p+1 \,..\, p+q \\ & \displaystyle\sum_{j=1}^{d} a_{ij} x_j \ge b_i & \text{for each } i = p+q+1 \,..\, n \\ \hline \end{array} \Longrightarrow \begin{array}{|rl|}\hline \max & c \cdot x \\ \text{s.t.} & Ax = b \\ & x \ge 0 \\ \hline \end{array}$$

   (b) Describe precisely how to dualize a linear program written in slack form.

   (c) Describe precisely how to dualize a linear program written in general form.

   In all cases, keep the number of variables in the resulting linear program as small as possible.

2. Suppose you have a subroutine that can solve linear programs in polynomial time, but only if they are both feasible and bounded. Describe an algorithm that solves *arbitrary* linear programs in polynomial time. Your algorithm should return an optimal solution if one exists; if no optimum exists, your algorithm should report that the input instance is UNBOUNDED or INFEASIBLE, whichever is appropriate. *[Hint: Add one variable and one constraint.]*

3. An *integer program* is a linear program with the additional constraint that the variables must take only integer values.

   (a) Prove that deciding whether an integer program has a feasible solution is NP-complete.

   (b) Prove that finding the optimal feasible solution to an integer program is NP-hard.

   [Hint: Almost any NP-hard decision problem can be formulated as an integer program. Pick your favorite.]

4. Give a linear-programming formulation of the *minimum-cost feasible circulation problem*. You are given a flow network whose edges have both capacities and costs, and your goal is to find a feasible circulation (flow with value 0) whose cost is as small as possible.

5. Given points $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ in the plane, the *linear regression problem* asks for real numbers a and b such that the line $y = ax + b$ fits the points as closely as possible, according to some criterion. The most common fit criterion is minimizing the L_2 *error*, defined as follows:[1]

$$\varepsilon_2(a, b) = \sum_{i=1}^{n} (y_i - ax_i - b)^2.$$

But there are several other fit criteria, some of which can be optimized via linear programming.

(a) The L_1 *error* (or *total absolute deviation*) of the line $y = ax + b$ is defined as follows:

$$\varepsilon_1(a, b) = \sum_{i=1}^{n} \left| y_i - ax_i - b \right|.$$

Describe a linear program whose solution (a, b) describes the line with minimum L_1 error.

(b) The L_∞ *error* (or *maximum absolute deviation*) of the line $y = ax + b$ is defined as follows:

$$\varepsilon_\infty(a, b) = \max_{i=1}^{n} \left| y_i - ax_i - b \right|.$$

Describe a linear program whose solution (a, b) describes the line with minimum L_∞ error.

[1] This measure is also known as *sum of squared residuals*, and the algorithm to compute the best fit is normally called *(ordinary/linear) least squares fitting*.