

Homework 3

Computer Science 511

Fall 2009

Due on Friday, October 2

Submission Instructions

This assignment must be submitted by 5 PM of its due date at the CS 511 drop box.

Reading Assignment

The two main references are the following.

- T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd or 3rd edition, MIT Press, 2001/2009 (Chapter 29).
- S. Dasgupta, C. Papadimitriou, and U. Vazirani, *Algorithms*, McGraw-Hill, 2008 (Chapter 7). An e-version of this reference is available at www.cs.berkeley.edu/~vazirani/algorithms/chap7.pdf.

Problem Set

1. (10 points) The Canine Products company offers two dog foods, Frisky Pup and Husky Hound, that are made from a blend of cereal and meat. A package of Frisky Pup requires 1 pound of cereal and 1.5 pounds of meat, and sells for \$7. A package of Husky Hound uses 2 pounds of cereal and 1 pound of meat, and sells for \$6. Raw cereal costs \$1 per pound and raw meat costs \$2 per pound. It also costs \$1.40 to package the Frisky Pup and \$0.60 to package the Husky Hound. A total of 240,000 pounds of cereal and 180,000 pounds of meat are available each month. The only production bottleneck is that the factory can only package 110,000 bags of Frisky Pup per month. Needless to say, management would like to maximize profit.
 - (a) Formulate the problem as a linear program in two variables.

- (b) Graph the feasible region, give the coordinates of every vertex, and circle the vertex maximizing profit. What is the maximum profit possible?
2. In the *minimum-cost multicommodity flow problem*, we are given a directed graph $G = (V, E)$ in which every edge $e \in E$ has a capacity $c(e) \geq 0$ and a cost $a(e)$. As in the multicommodity flow problem, we are given k different commodities K_1, K_2, \dots, K_k , where commodity i is specified by the triple $K_i = (s_i, t_i, d_i)$, in which s_i, t_i , and d_i denote the source, destination and demand for commodity i . We write f_i to denote the flow for commodity i . For each edge $e \in E$, the aggregate flow on e , denoted $f(e)$, is the sum of all flows on e ; that is, $f(e) = \sum_{i=1}^k f_i(e)$. A feasible flow is defined in the same way as for the multicommodity flow problem; thus, the aggregate flow on each edge cannot exceed the capacity of that edge, and the individual flows must satisfy flow conservation and meet the demand for each commodity. The cost of a flow is $\sum_{e \in E} a(e)f(e)$, and the goal is to find a feasible flow of minimum cost. Express this problem as a linear program.
3. Consider the following linear program:

$$\begin{aligned} & \text{maximize} && x - y \\ & \text{subject to} && ax + by \leq -1 \\ & && x, y \geq 0, \end{aligned} \tag{1}$$

where a and b are real numbers.

- (a) Under what conditions on the values of a and b is linear program (1) infeasible? Justify your answer.
- (b) Under what conditions on the values of a and b is linear program (1) unbounded? Justify your answer.
- (c) Under what conditions on the values of a and b does linear program (1) have a finite and unique optimal solution? Justify your answer.
4. Solve the following linear program using the simplex algorithm.

$$\begin{aligned} & \text{minimize} && x_1 + x_2 + x_3 \\ & \text{subject to} && 2x_1 + 7.5x_2 + 3x_3 \geq 10000 \\ & && 20x_1 + 5x_2 + 10x_3 \geq 30000 \\ & && x_1, x_2, x_3 \geq 0 \end{aligned}$$

5. Given a set of m linear inequalities on n variables x_1, x_2, \dots, x_n , the *linear-inequality feasibility problem* asks if there exists a setting of the variables that simultaneously satisfies each of the inequalities.

- (a) Show that if we have an algorithm for linear programming, we can use it to solve the linear-inequality feasibility problem. The number of variables and constraints that you use in the linear-programming problem should be polynomial in n and m .
- (b) Show that if we have an algorithm for the linear-inequality feasibility problem, we can use it solve the linear programming programming problem. The number of variables and inequalities that you use in the linear-inequality feasibility problem should be polynomial in n and m , the number of variables and constraints in the linear program.
6. An *integer linear program* is a linear programming problem with the additional constraint that the variables x must take on integral values. It turns out that just determining whether an integer linear program has a feasible solution is NP-hard, which means that it is unlikely that there is a polynomial-time algorithm for this problem.
- (a) Show that weak duality holds for an integer linear program.
- (b) Show that duality does not always hold for an integer linear program.
- (c) Given a primal linear program in standard form, let us define P to be the optimal objective value for the primal linear program, D to be the optimal objective value for the dual, IP to be the optimal objective value for the integer version of the primal (that is, the primal with the added constraint that the variables take on integer values), and ID be the optimal objective value for the integer version of the dual. Show that

$$IP \leq P = D \leq ID.$$

Note. We reserve the right to grade only a subset of the problems assigned. Which problems will be graded will be decided after the submission deadline.