CS358: Applied Algorithms

Assignment 6: Integer Linear Programming (due 5/14/2020 10PM EDT)

Instructor: Sam McCauley

Instructions

All submissions are to be done through github. This process is detailed in the handout "Handing In Assignments" on the course website. Answers to the questions below should be submitted by editing this document. All places where you are expected to fill in solution are marked in comments with "FILL IN."

Please contact me at sam@cs.williams.edu if you have any questions or find any problems with the assignment materials.

This assignment does not have a programming component. Your answers to these questions will determine your grade.

Questions

Problem 1 (10 points)**.** In the lecture we discussed an ILP for the Two Towers problem (from Assignment 1). Consider solving this problem by relaxing the integer constraints, treating the problem as an LP, and then rounding to find the final solution. Is this likely to give a good Two Towers solution? Briefly explain your answer.

Solution.

Problem 2 (10 points)**.** You have a problem that is nearly an LP:

Objective:

$$\max x + y$$

Constraints:

$$\begin{aligned} x^2 + y^2 &\leq 9 \\ x - y &\leq 2 \\ x &\geq 0 \\ y &\geq 0 \end{aligned}$$

Will the simplex algorithm give a correct solution to this problem? Please explain briefly why it always will, or why it may not.

Hint: Drawing a picture of the search space may help.

Solution.

Problem 3 (40 points)**.** Consider a variant of the two towers problem where we use *rectangular* (rather than square) blocks:

You are given a set of n pairs $(\ell_1, w_1), (\ell_2, w_2), \dots, (\ell_n, w_n)$, where ℓ_i and w_i represent the length and width of block i respectively.

Your job is to create two stacks of blocks of height as equal as possible. This assignment must specify which tower the block lies in, as well as how the block is oriented. That is to say, your goal is to partition all n blocks into four sets L_1, W_1, L_2, W_2:

- L_1 is the set of blocks in tower 1 with length oriented vertically,
- W_1 is the set of blocks in tower 1 with width oriented vertically,
- L_2 is the set of blocks in tower 2 with length oriented vertically,
- W_2 is the set of blocks in tower 2 with width oriented vertically.

Then your goal is to minimize the difference between the heights of towers 1 and 2; that is to say, your goal is to minimize:

$$\left| \left(\sum_{i \in L_1} \ell_i + \sum_{i \in W_1} w_i \right) - \left(\sum_{i \in L_2} \ell_i + \sum_{i \in W_2} w_i \right) \right|$$

Give an MIP to solve this problem.

Solution.

Problem 4 (40 points)**.** You've been asked to help schedule exams at Williams College. There are n students and m courses. You are told which student is taking which course with nm indicator variables: $c_{i,j} = 1$ if student i is taking course j, and $c_{i,j} = 0$ otherwise. You may assume that each student is taking exactly four courses.

Your job is to assign each course to a time slot such that if a student i is taking two courses j_1 and j_2 (that is to say, if $c_{i,j_1} = 1$ and $c_{i,j_2} = 1$), then courses j_1 and j_2 are assigned to two different time slots.

(a) Design an MIP to minimize the number of slots k necessary to schedule all courses.

(b) There may be many schedules that use only k slots. Among these, we would like to find a schedule that minimizes the number of times a student needs to take two consecutive exams. (That is to say, the MIP should minimize the number of pairs (s, t) where student s has an exam in time slot t, and also has an exam in time slot $t + 1$.) This question is split into two parts:

(i) We want a variable $e_{i,j} \in \{0, 1\}$ to help keep track of this new cost. Design MIP constraints to ensure that $e_{i,j} = 1$ if student i has an exam in time slot j, and also has an exam in time slot $j + 1$.

(ii) Modify the objective function using $e_{i,j}$ so that the MIP solution minimizes the number of slots k, breaking ties by minimizing the number of times a student needs to take consecutive exams. *Hint:* The number of slots is much more important than the number of students with exams in consecutive slots. Is there a number you can multiply k by to ensure that minimizing k is more important?

Solution.