CS 170 Homework 11

Due **Monday 4/6/2026, at 10:00 pm (grace period until 11:59pm)**

Study Group

List the names and SIDs of the members in your study group. If you have no collaborators, explicitly write “none”.

1 Making Change (Solo Question; 10 points)

You are using a currency with coins of n different positive values $v_1, \ldots, v_n$. You want to determine the minimum possible number of coins needed to make change for a given value m, or else determine that it's impossible. You may use multiple coins of the same value v_i.

For example, if $m = 10$ and there are $n = 3$ coins with values $v_1 = 2$, $v_2 = 4$, $v_3 = 8$, then the optimal solution uses two coins, as $10 = 2 + 8$.

(a) Formulate this problem as an integer linear program, meaning that on top of the linear constraints, we require every variable to be an integer. Note that the values v_i are fixed, and hence will not be variables.

(b) Can we solve this problem as an ordinary linear program, and be sure that the solution will only use integer numbers of coins?

2 Weighted Rock-Paper-Scissors (10 points)

You and your friend used to play rock-paper-scissors, and have the loser pay the winner 1 dollar. However, you then learned in CS170 that the best strategy is to pick each move uniformly at random, which took all the fun out of the game.

Your friend, trying to make the game interesting again, suggests playing the following variant: If you win by beating rock with paper, your opponent pays you 5 dollars. If you win by beating scissors with rock, they pay you 3 dollars. If you win by beating paper with scissors, they pay you 1 dollar.

(a) Draw the payoff matrix for this game.

(b) Write a linear program to find the optimal strategy.

(c) Can you determine your optimal expected payoff, without solving the linear program?

Hint: Compare the primal and dual optimization problems.

3 Domination (10 points)

In this problem, we explore a concept called *dominated strategies*. Consider a zero-sum game with the following payoff matrix for the row player:

		Column:		
		A	B	C
	D	1	2	-3
Row:	E	3	2	-2
	F	-1	-2	2

Recall that the row player wants to maximize the value, and the column player wants to minimize the value.

(a) If the row player plays optimally, can you find the probability that they pick D without directly solving for the optimal strategy? Justify your answer.

(b) Given the answer to part (a), if both players play optimally, what is the probability that the column player picks A?

(c) Given the answers to part (a) and (b), what are both players' optimal strategies?

Note: All parts of this problem can be solved without using an LP solver or solving a system of linear equations.

4 Shortest Path (15 points)

In this question, we will see a linear program for the shortest path problem between fixed vertices s and t in a directed graph $G = (V, E)$. Assume that each edge e has edge weight w_e, such that there are no negative-weight cycles.

We define a variable $x_e \geq 0$ for each edge e, which represents a flow along edge e. As usual, the incoming flow to each vertex $v \neq s, t$ equals the outgoing flow, that is,

$$\sum_{e=(u,v)\in E} x_e - \sum_{e=(v,u)\in E} x_e = 0.$$

We require one total unit of flow from s to t, meaning we also have

$$-\sum_{e=(s,u)\in E} x_e = -1$$
$$\sum_{e=(u,t)\in E} x_e = 1,$$

where each expression above sums over vertices u forming the specified edges.

(a) Write a linear objective function whose solution under these constraints gives the length of the shortest path from s to t. Briefly explain your reasoning.

(b) Write down the dual LP.

(c) An interpretation of the dual is given on page 210 of DPV. Why isn't our dual LP identical to the one given there?

5 Slowing Down Simplex (15 points)

In this question, we'll visualize a "worst-case example" (similar to Klee-Minty) of a linear program for which the simplex algorithm can visit all vertices of the feasible region before finding the optimum. Recall that the simplex algorithm repeatedly walks along edges in a direction that increases the objective value.

(a) Suppose we have two variables x, y, and our feasible region is the unit square $0 \leq x, y \leq 1$. Our objective is to maximize y. Show how to perturb the constraints so that there is a path along the edges of the feasible region that starts at the origin and visits every vertex, such that the objective value y strictly increases at each step.

State your new constraints, and draw the new feasible region along with the desired path.

Hint: You want to allow the walk to first traverse the bottom edge, then move up, and then traverse the top edge. Can you update the lower and upper bounds on y to "tilt" the edges so that y is always increasing during this walk?

(b) Now suppose we have three variables x, y, z, our feasible region is the unit cube $0 \leq x, y, z \leq 1$, and our objective is to maximize z. Perturb the constraints so that there is a path along the edges of the feasible region that starts at the origin and visits every vertex, such that z strictly increases at each step.

Specifically, state your new constraints, and draw the desired path.

Hint: Now update the bounds on z to tilt the bottom and top faces. Within each face, use part (a).

(c) (Optional, not for credit) Can you see how to generalize this example to perturb the n-dimensional hypercube $0 \leq x_1, \ldots, x_n \leq 1$ when $n > 3$? We want to ensure that the simplex algorithm can visit all 2^n vertices, hence taking exponential time.