15-451 Algorithms, Spring 2017
Recitation #8 Worksheet

A. Rock-Paper-Scissors with a Twist Suppose we have a non-standard game of rock-paper-scissors, which is still zero-sum, but with the following payoffs for the row player (Alice):

<div align="center">

Bob plays

Alice plays		r	p	s
	r	0	-1	2
	p	1	0.5	-1
	s	-1	2	-1

</div>

1. If Alice decides to play $\mathbf{p} = (p_1, p_2, 1 - p_1 - p_2)$ as her strategy, what should Bob play to minimize the payoff to Alice? What is Alice's payoff if he does this.

   Solution: Bob can either play rock, or paper, or scissors. He'll play whichever gives the lowest payoff to Alice. All other mixed strategies are averages of these three pure strategies. In this case Alice's payoff will be

$$\min \left\{ \begin{array}{lll} p_1 \cdot 0+ & p_2 \cdot 1+ & (1 - p_1 - p_2) \cdot (-1), \\ p_1 \cdot (-1) + & p_2 \cdot (0.5) + & (1 - p_1 - p_2) \cdot (2), \\ p_1 \cdot (2)+ & p_2 \cdot (-1)+ & (1 - p_1 - p_2) \cdot (-1) \end{array} \right\}$$

2. Hence, write down the linear program that Alice must solve in order to find her best strategy $\mathbf{p}^\star$. You don't have to solve for the value of this game.

   Solution: Alice will try to maximize her payoff, given Bob plays the strategy in part B. So she will try to solve

$$\max v$$
$$\text{subject to} \quad p_1 \cdot 0 + p_2 \cdot 1 + (1 - p_1 - p_2) \cdot (-1) \geq v$$
$$p_1 \cdot (-1) + p_2 \cdot (0.5) + (1 - p_1 - p_2) \cdot (2) \geq v$$
$$p_1 \cdot (2) + p_2 \cdot (-1) + (1 - p_1 - p_2) \cdot (-1) \geq v$$
$$p_1 + p_2 \leq 1$$
$$p_1 \geq 0, p_2 \geq 0$$

   Note that taking the new variable v and making all the three expressions at least as large as v, and then maximizing v makes sure that v is the minimum of the three expressions as desired in the previous part.

B. Taking Duals I. Consider this maximization linear program:

$$\max(x_1 + 3x_2 - 2x_3)$$
$$\text{s.t.} \quad x_1 + x_2 + 2x_3 \leq 2$$
$$7x_1 + 2x_2 + 5x_3 \leq 6$$
$$2x_1 + x_2 - x_3 \leq 1$$
$$x_1, x_2, x_3 \geq 0$$

1. Write down its dual LP. (Is it a maximization or minimization problem? What are the variables? Constraints?)

   Solution: Remember, we're trying to find the best (i.e., lowest) upper bound on the value of the primal LP. So it should be a *minimization*. There's one variable y_i for each constraint in the primal, so that's *three y_i variables*. We're summing up constraints of the form $blah \geq blah$, so we want each $y_i \geq 0$ to not flip the inequalities. Etc.

$$\min(2y_1 + 6y_2 + 1y_3)$$
$$\text{s.t.} \quad y_1 + 7 \ y_2 + 2 \ y_3 \geq 1$$
$$y_1 + 2 \ y_2 + 1 \ y_3 \geq 3$$
$$2 \ y_1 + 5 \ y_2 + (-1) \ y_3 \geq -2$$
$$y_1, y_2, y_3 \geq 0$$

2. Now write down the dual of this dual LP. Remember, since the dual will be a minimization LP, this dual's-dual will give a best lower bound on the dual.

   Solution: The dual's dual will be the same as the primal. Details are omitted here.

Taking Duals II. Write down the dual of this minimization LP: *(Be careful, some inequalities are greater-than, some are less-than. And not all constraints have all variables.)*

$$\max(x_1 - 3x_2 + 2x_3)$$
$$\text{s.t.} \quad 3x_1 + 2x_3 \geq 2$$
$$2x_2 - x_3 \leq 5$$
$$x_1, x_2, x_3 \geq 0$$

Solution: To make life easy, first convert this into a nicer form. Since the dual wants to give a *upper bound* on this *maximization* problem, let's make the constraints of the form $blah \leq blah$.

$$\max(x_1 - 3x_2 + 2x_3)$$
$$\text{s.t.} \quad {\color{red}-3x_1 - 2x_3 \leq -2}$$
$$2x_2 - x_3 \leq 5$$
$$x_1, x_2, x_3 \geq 0$$

And not all variables appear in all constraints, so let's put down zeros where things are missing.

$$\max(x_1 - 3x_2 + 2x_3)$$
$$\text{s.t.} \quad -3x_1 \textcolor{red}{+0x_2} - 2x_3 - 2$$
$$\textcolor{red}{0x_1} + 2x_2 - x_3 \leq 5$$
$$x_1, x_2, x_3 \geq 0$$

Finally, use a dual variable $y_i \geq 0$ for each primal constraint, and two constraints, etc.

$$\min(-2y_1 + 5y_2)$$
$$\text{s.t.} \quad -3y_1 + 0y_2 \geq 1$$
$$0y_1 + 2y_2 \geq -3$$
$$-2y_1 - y_2 \geq 2$$
$$x_1, x_2, x_3 \geq 0$$

C. Minimax from Duality (by Example). Let the row-player's payoffs be given by this (non-negative) matrix

	L	R
L	1	5
R	3	2

1. If the probabilities on the two rows are p_1 and p_2, write down an LP for the row player's optimal strategy:

   Solution: If the row player puts $p_1 \geq 0$ and $p_2 \geq 0$ on L, R respectively, then she wants to solve $\max_{p_1+p_2=1, p_1, p_2 \geq 0} \min(p_1 + 3p_2, 5p_1 + 2p_2)$. I.e., the LP is

   $$\max v$$
   $$\text{subject to} \quad p_1 + 3p_2 \geq v$$
   $$5p_1 + 2p_2 \geq v$$
   $$p_1 + p_2 \leq 1$$
   $$p_1, p_2 \geq 0.$$

   We set $p_1 + p_2 \leq 1$, but to maximize v, the LP will automatically set the sum *equal to* 1.

2. Now take the dual of this LP. Show this dual is an LP computing the column player's optimal strategy. (And hence strong duality implies the minimax theorem.)

   Solution: First convert into inequalities $blah \leq blah$ useful to show an upper bound. (Also, since all payoffs are non-negative, we can add in non-negativity for v.

   $$\max v$$
   $$\text{subject to} \quad v - p_1 - 3p_2 \leq 0$$
   $$v - 5p_1 - 2p_2 \leq 0$$
   $$p_1 + p_2 \leq 1$$
   $$v, p_1, p_2 \geq 0.$$

If the dual variables are q_1, q_2 and w, we get

$$\min\ w$$
$$\text{subject to}\quad q_1 + q_2 \geq 1$$
$$-q_1 - 5q_2 + w \geq 0$$
$$-3q_1 - 2q_2 + w \geq 0$$
$$w, q_1, q_2 \geq 0$$

Now move some variables around:

$$\min\ w$$
$$\text{subject to}\quad q_1 + q_2 \geq 1$$
$$w \geq q_1 + 5q_2$$
$$w \geq 3q_1 + 2q_2$$
$$w, q_1, q_2 \geq 0$$

And again observe that to minimize the value, any optimal solution will reduce q_1, q_2 to make their sum equal to 1. So the LP is solving:

$$\min_{q_1+q_2=1, q_1, q_2 \geq 0}\ \max(q_1 + 5q_2, 3q_1 + 2q_2).$$

That's the column player's strategy!!! And by strong duality, we get the minimax theorem for this particular game. Exactly the same idea holds in general, details are in the lecture notes.

D. Coding up shortest paths as an LP: In lecture you saw that you can code up the s-t shortest-path problem as an LP. The input is a directed graph G with edge weights $w(e) \geq 0$, start node s, and a target t. We want to find a path from s to t of least weight.

1. *What are the variables?* Suppose we have a variable d_v for every vertex v, representing its distance from s. What are the constraints you should write?

   Solution: For any edge $e = (u, v)$ we put the constraint that $d_v \leq d_u + w(e)$. In other words, since one way to reach v is to reach u first and then go on edge e to v, the shortest-path distances must satisfy this property. We also add the constraint $d_s = 0$.

2. *The objective?* We claim it is *maximize d_t*. Why does this make sense?

   Solution: Any solution will have values d_v that are *no larger* than the shortest path from s to v, since by induction (on nodes in order of their true distance from s) the node u that comes right before v in the true shortest path from s will never have too large a value, and we have constrained $d_v \leq d_u + w(u, v)$. So d_t cannot be more than the shortest s-t distance.

   Also, if you set d_v to be the actual shortest-path distance from s to v, then this satisfies all the constraints in the LP. So the optimal value of d_t cannot be less than the shortest s-t distance either. Hence it must be equal.

Another way to think about this: take a set of stones (one for each node) and strings (one for each edge), and tie a string of length $w(u, v)$ between stones u and v. Now pull s and t as far as possible from each other: the strings that become taut are those on the shortest path from s to t, and the furthest s and t can be pulled from each other equals their shortest-path distance.

D'. Coding up shortest paths as an LP (Approach II): Have a variable x_e for each edge e, with constraints that $0 \leq x_e \leq 1$. (Think of $x_e = 1$ meaning we use that edge, and $x_e = 0$ meaning we don't, but of course the LP might assign fractional values.) Our goal is to minimize $\sum_e w(e) x_e$, subject to:

- One unit of "flow" leaves s: $\sum_{e=(s,v)} x_{sv} = 1$
- One unit of "flow" enters t: $\sum_{e=(v,t)} x_{vt} = 1$.
- For all $v \notin \{s, t\}$, we have flow-in = flow-out: $\sum_{e=(u,v)} x_{uv} = \sum_{e=(v,u)} x_{vu}$.

This is a min-cost flow (send 1 unit of s-t-flow with least cost). You can put edge-capacity 1, but since we send only 1 unit of flow, the capacity constraints don't matter.

1. Solve the LP to get an optimal solution. If we get back an integer solution (i.e., if $x_e \in \{0, 1\}$ for each edge e) argue that the edges with $x_e = 1$ give a shortest s-t path.

   Solution: The flow gives a path. And if there is a shorter path, that would be a solution with smaller cost to the LP as well, contradicting the optimality of x.

2. Suppose the optimal LP solution returns a fractional flow. Argue that any flow-carrying path from s to t is a shortest path.

   Solution: Take the flow f and decompose it into flows on s-t paths, say α_i flow on path P_i, with $\sum_i \alpha_i = 1$. The LP objective function is $\sum_i \alpha w(P_i)$, where $w(P_i) = \sum_{e \in P_i} w_e$. Now if some P_i with $\alpha_i > 0$ was not a shortest path, we could reduce flow on that and put more flow on other paths to get a smaller cost LP solution.

Now, what if the LP solver returns fractional values? The claim is that in that case, you can get at least as good an LP solution by rerouting that flow on the shortest path. So there exists an optimal integer solution. (And a vertex solution, that simplex will find, will be integral.) We got lucky here that the fractional values didn't hurt us. For other problems (e.g., the non-bipartite matching problem, or 3SAT and vertex-cover that we'll see in the coming lectures) you can't necessarily convert a fractional solution into an integer one.