In order to prepare this submission, the following resource(s) has been used:

- Discussion with fellow students: *Victor Gan*, *Sarah Huber*, and *Evan Peng*.

1. (from DPV Problem 7.28) A linear program for shortest path. Suppose we want to compute the shortest path from node s to node t in a directed graph with edge lengths $l_e > 0$.

   (a) Show that this is equivalent to finding a s-t flow f that minimizes $\sum l_e f_e$ subject to $size(f) = 1$. There are no capacity constraints.
   Lemma 1. If there exists a s-t flow f that minimizes $\sum l_e f_e$ subject to $size(f) = 1$, then there exists an equivalent flow f' which consists of only one path from s to t.
   Lemma 1 Proof. If the flow f is already a single path, we are done. Otherwise, Let's select a shortest path p_s, and then pick a random path p_r from s to t in our flow. We select Q_p to be the minimum $f_{e \in p_r}$. Given the linearity of the objective function, Q_p could be considered as a flow which has gone from s to t with partial penalty $\sum_{e \in p_r} l_e Q_p = Q_p \sum_{e \in p_r} l_e$. $\sum_{e \in p_r} l_e$ can't be greater than $\sum_{e \in p_s} l_e$, which is the length for the shortest path, otherwise we can direct the flow Q_p to p_s and lower the overall objective which will contradict the first assumption of having a flow satisfying the mentioned conditions. It also can't be smaller because that will contradict the assumption that p_s is a shortest path. Thus we can direct the flow Q_p from p_r to p_s without losing anything. Since Q_p was the smallest flow in path P_r, changing the flow path will eliminate at least one edge from the f. Since the number of edges are finite, after repeating this process we will end up with a single path (and flow f') from s to t which still satisfies the conditions and is not worse than the flow f.
   Problem Proof. First I prove that finding a *s-t* flow f satisfying the above conditions implies finding a shortest path from s to t. Given the Lemma 1, any f satisfying these constraints will have an equivalent flow f which is only a single path p from s to t. Thus the flow along that path will be 1, because the flow size is 1. Thus the objective function will become $min(\sum_{e \in p} l_e)$, which in conjunction with p being a path from s to t, is the definition for the shortest path from s to t. For the other way, let's assume the shortest path is p. If p is the flow satisfying the constraints, we're done. Otherwise, let's assume the flow f is the best answer to the problem. Considering the Lemma 1, there exists a single path p' which also satisfies this condition and forms the flow f' which is equivalent to flow f. Given the assumptions $length(p')$ must be less than or equal to $length(p)$, because p is the shortest path. It also can't be longer than p, because then p will yield a better flow f'' which contradicts the initial assumption. Thus the shortest path must be a single path answer to the above optimization problem. So these two problems are equivalent.

   (b) Write the shortest path problem as a linear program.
   Answer. We simply have to convert the above flow problem into a linear program.

$$\text{Minimize } Z = \sum l_e f_{uv} \text{ subject to:} \tag{1}$$

$$\sum_{v \in V} f_{sv} = \sum_{v \in V} f_{vt} = 1 \text{ (Define s and t property)} \tag{2}$$

$$\text{For each } u \in V : \sum_{v \in V} f_{uv} = \sum_{v \in V} f_{vu} \text{ (Flow conservation)} \tag{3}$$

$$f_{uv} \geq 0 \tag{4}$$

(c) Show that the dual LP can be written as:

$$\max x_s - x_t$$
$$x_u - x_v \leq l_{uv} \text{ for all } (u,v) \in E$$

Answer. Expanding the linear program we'd have:

$$\text{Minimize } Z = \sum l_e f_{uv} \text{ subject to:} \tag{5}$$
$$\sum_{v \in V} f_{sv} = 1 \tag{6}$$
$$\sum_{v \in V} f_{vt} = 1 \tag{7}$$
$$\text{For each } u \in V : \sum_{v \in V} f_{uv} - \sum_{v \in V} f_{vu} = 0 \tag{8}$$

By multiplying 6 by X_s and 7 by $-X_t$ and 8 by X_u and then summing all the equalities, we'd get:

$$\text{Minimize } Z = \sum l_e f_{uv} \text{ subject to:} \tag{9}$$
$$X_s \sum_{v \in V} f_{sv} = X_s \tag{10}$$
$$-X_t \sum_{v \in V} f_{vt} = -X_t \tag{11}$$
$$\text{For each } u \in V : X_u \sum_{v \in V} f_{uv} - X_u \sum_{v \in V} f_{vu} = 0 \tag{12}$$
$$\Rightarrow \sum_{(u,v) \in E} f_{uv}(X_u - X_v) = X_s - X_t \tag{13}$$

13 results from the fact that for each edge (u,v) we have exactly $-X_v$ and $+X_u$ in the other equalities, because each edge appears exactly once in the entrance of vertex v $(-X_v)$, and on the departure from one vertex u $(+X_u)$.
As long as $X_u - X_v \leq l_{uv}$, the function $\sum_{(u,v) \in E} f_{uv}(X_u - X_v)$ will be less than or equal to the objective function $\sum_{(u,v) \in E} f_{uv} l_{uv}$ (Note that $f_{uv} \geq 0$). So this function (subject to constraint $X_u - X_v \leq l_{uv}$) bounds the minimum value for the objective function $\sum_{(u,v) \in E} f_{uv} l_{uv}$. So by maximizing the lower bound for the objective function, we'd be able to identify the minimum value for the objective function. Luckily, due to the 13, this bounding function is also equal to $X_s - X_t$, thus it suffices to maximize $X_s - X_t$ subject to the constraint $X_u - X_v \leq l_{uv}$. So the dual LP can be formulated as:

$$\max X_s - X_t$$
$$X_u - X_v \leq l_{uv} \text{ for all } (u,v) \in E$$

2. Let G be a bipartite graph. We want to find a minimum edge cover; show how to solve this problem using a maximum flow algorithm.

Answer. First, we add a directed edge from S to vertices in partition 1. For each vertex v that we add an edge to, we set the flow capacity to be equal to $c(S, v) = |n(v)| - 1$, where $n(v)$ is the set of neighbors of v. Then, from each vertex u in partition 2, we add a directed edge to T with flow capacity of $c(u, T) = |n(u)| - 1$. Finally, we set the capacity of edges between partition 1 and 2 to be 1.

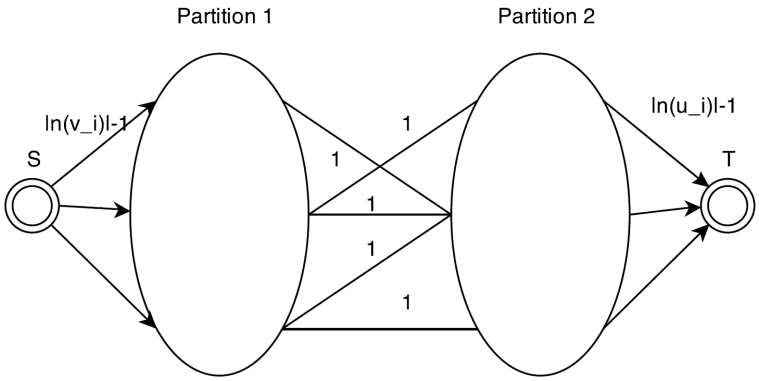

Figure 1: Converting the edge-cover bipartite graph problem to a network flow problem.

Now I claim the maximum flow in this graph yields the minimum edge-cover. The edges that are not actually part of the flow, are the edges we need for the covering problem. **Proof.**

(a) For each possible flow in this graph, there's an equivalent edge cover solution.
Proof. Since each arc going into a vertex v, has the capacity of $|n(v)| - 1$, in any flow there always exists an edge from v to partition 2 that's not used (otherwise it'll be breaking the capacity constraint of the entering edge). The same argument goes for the second partition; in any flow from u there exists at least an edge coming from partition 1 which is not used. Therefore, all the unused edges in this flow will be creating an edge cover.

(b) For any edge cover, there exists at least an equivalent flow.
Proof. Simply remove the edges of the cover from the original graph and then solve the max flow problem.

(c) The equivalent edge cover of any flow from S to T has the size of

$$|E| - size(f) \tag{14}$$

where $|E|$ is the number of edges.
Proof. Given that

$$size(f) = \sum_{v_i \in partition1} f(S, v_i) \tag{15}$$

$$\forall_{v_i \in partition1} f(S, v_i) \leq c(S, v_i) < |n(v_i)| \tag{16}$$

The number of edges that are not involved in the flow will be the total number of edges minus flow size.

(d) **Conclusion.** In order to minimize the number of edges for the edge cover problem, we have to minimize the 14, which is equivalent to maximizing the $size(f)$ which leads us to the

max flow problem. Thus solving the max-flow problem will give us the minimum edge cover solution.

3. Each of two players hides a nickel (5 cents) or a dime (10 cents). If the two coins match then A gets both; if they don't match then B gets both.

   (a) Show the payoff matrix. (The payoff is the additional money a player gets.)
   Answer. For A:

	5	10
5	+5	-5
10	-10	+10

   (b) Describe the optimal strategy for A using a linear program. What is the optimal strategy?
   Answer. Let's say for A: $P(X = 5) = x$ and $P(X = 10) = y$. For B, the options will be $5x - 10y$ and $-5x + 10y$. Obviously, B will be trying to minimize the loss; and A must maximize the minimum loss for A to guarantee the gain. The following constraints will show us the solution.

$$\text{Maximize } P = z + 0x + 0y \text{ subject to} \tag{17}$$
$$z - 5x + 10y <= 0 \tag{18}$$
$$z + 5x - 10y <= 0 \tag{19}$$
$$x + y = 1 \tag{20}$$
$$0 <= x, y <= 1 \tag{21}$$
$$\Rightarrow x = \frac{2}{3}, y = \frac{1}{3}, p = 0 \tag{22}$$

   So player A must pick 5 with probability 2/3, and 10 with probability 1/3; and doing so, he can guarantee to not lose anything.

   (c) What is the optimal strategy if instead of a nickel or a dime the players hide a c-cent coin or a d-cent coin?
   Answer.

$$\text{Maximize } P = z + 0x + 0y \text{ subject to} \tag{23}$$
$$z - cx + dy <= 0 \tag{24}$$
$$z + cx - dy <= 0 \tag{25}$$
$$x + y = 1 \tag{26}$$
$$0 <= x, y <= 1 \tag{27}$$
$$24, 25 \Rightarrow z <= 0 \tag{28}$$
$$z = 0 \Rightarrow z = 0 = cx - dy \Rightarrow \frac{c}{d}x = y \tag{29}$$
$$26, 29 => \frac{c}{d}x = 1 - x \Rightarrow x = \frac{d}{c+d}, y = \frac{c}{c+d}. \tag{30}$$

   So we must pick c with probability $\frac{d}{c+d}$, and d with probability $\frac{c}{c+d}$; and doing so will guarantee no loss.