CS 1360 Spring 2025
Midterm Exam

Name: ______________________________

Harvard ID: ___________________________

Name: ______________________________

Problem 1: Equilibrium Computation

1. **[10 pts]** Define (using mathematical notation) the notion of *maximin strategy* (of player 1) in a two-player zero-sum game.

   Solution: The Maximin (randomized) strategy of player 1 is

   $$x_1^\star \in \arg \max_{x_1 \in \Delta(S_1)} \min_{s_2 \in S_2} u_1(x_1, s_2)$$

   The Minimax (randomized) strategy of player 2 is

   $$x_2^\star \in \arg \min_{x_2 \in \Delta(S_2)} \max_{s_1 \in S_1} u_1(s_1, x_2)$$

2. **[15 pts]** Write down a linear program that computes a maximin strategy in a two-player zero-sum game.

   Note: This was done in class.

   Solution: The Maximin strategy is computed via LP (and the minimax strategy is computed analogously):

   $$\begin{aligned} \max \quad & w \\ \text{s.t.} \quad & \forall s_2 \in S, \quad \sum_{s_1 \in S} p(s_1) u_1(s_1, s_2) \geq w, \\ & \sum_{s_1 \in S} p(s_1) = 1, \\ & \forall s_1 \in S, \quad p(s_1) \geq 0. \end{aligned}$$

   Here, w is the maximum utility player 1 can achieve when player 2 minimizes player 1's utility. The first set of constraints says that every strategy player 2 plays must result in a utility for player 1 of at least w. The second and third constraints are just trivial constraints on the probabilities in player 1's mixed strategy.

Name: ______________________________

Problem 2: The Price of Anarchy

1. **[10 pts]** Describe in words how to establish an upper bound and a lower bound on the price of anarchy of a given class of games. As in class, the solution concept is pure Nash equilibrium and the objective function is social cost (sum of costs).

   Solution: To derive an upper bound of x, we need to show that for every game in the class and every equilibrium of the game, the ratio between the cost of the equilibrium and the optimal cost is at most x.

   The derive a lower bound of x, we need to give an example of a game in the class and an equilibrium of the game such that the ratio between the cost of the equilibrium and the optimal cost is at least x.

2. **[15 pts]** Consider the class of all 2-player normal-form games with costs that are integers in $\{1, 2, \ldots, k\}$, that is, for every (pure) strategy profile (s_1, s_2), the cost of player i is $c_i(s_1, s_2) \in \{1, 2, \ldots, k\}$. Once again, the solution concept is pure Nash equilibrium and the objective function is social cost. Show that the price of anarchy is exactly k.

   Hint: Establish an upper bound (immediate) and a lower bound (a 2×2 game suffices).

   Solution:

   Upper bound: the worst-case Nash equilibrium has a social cost of 2k (both players incur a cost of k). The optimal social cost is 2, if both players incur a cost of 1. Hence our upper bound is $\frac{2k}{2} = k$, i.e. PoA $\leq k$.

   Lower bound: to show that this bound is tight, we construct a specific game where the PoA reaches k. Consider the following 2×2 game:

	A_1	A_2
A_1	(k, k)	$(2, k)$
A_2	$(k, 2)$	$(1, 1)$

   The NE are $(A_1, A_1), (A_2, A_2)$. The former has a social cost of $2k$ and the latter a social cost of 2. Hence our lower bound is $2k/2 = k$, i.e. PoA $\geq k$.

Name: ____________________

Problem 3: Voting Rules

1. **[10 pts]** Describe in words the concepts *Condorcet winner* and *Condorcet consistency*.

   Solution: A **Condorcet winner** is an alternative that defeats every other alternative in a head-to-head comparison.

   A rule is **Condorcet consistent** if it always selects a Condorcet winner whenever it is presented with a profile that contains one.

2. **[15 pts]** In a given preference profile, we say that alternative x is a *majority winner* if a majority of voters (more than $n/2$) rank x first. We say that a voting rule (social choice function) is *majority consistent* if the rule selects the majority winner whenever it is given a preference profile in which such an alternative exists.

   Prove or disprove the following statements:

   - Any Condorcet consistent voting rule is majority consistent.
   - Any majority consistent voting rule is Condorcet consistent.

   Solution:

   Any Condorcet consistent voting rule is majority consistent. True. Suppose an alternative x is a majority winner, where a majority of voters rank x first. Then, for a majority of voter's preference voters, x will defeat all other alternatives in a head to head comparison. By definition, x is also a Condorcet winner. This means that any majority winner is also a Condorcet winner, so any Condorcet consistent voting rule is majority consistent.

   Any majority consistent voting rule is Condorcet consistent. False. Proof by counterexample: consider the plurality rule. It is majority consistent, since plurality only considers each voter's top choice. When a majority of voters rank an alternative x first, x wins. However, plurality is not Condorcet consistent. Consider plurality among the following ranking preference profile:

1	2	3	4	5
a	a	b	c	d
b	b	c	b	b
c	c	d	d	c
d	d	a	a	a

Here b is the Condorcet winner, but plurality selects a.

Name: ________________________________

Problem 4: Strategic Manipulation in Elections

1. **[10 pts]** Define the f-Manipulation problem for a voting rule f.

   Solution: The $f - MANIPULATION$ problem is defined as follows: Given votes of non-manipulators and a preferred alternative p, can a manipulator cast a vote that makes p uniquely win under f?

2. **[15 pts]** Let us define another computational problem, the f-Shmanipulation problem. In this problem, we are given a voter $i \in N$ and a complete preference profile $\boldsymbol{\sigma}$, and we are asked whether there is a ranking σ_i' such that $f(\sigma_i', \boldsymbol{\sigma}_{-i}) \succ_{\sigma_i} f(\boldsymbol{\sigma})$.

   Show that if the voting rule f is such that there is a polynomial-time algorithm for f-Manipulation, then there is a polynomial-time algorithm for f-Shmanipulation.

   Solution: We can construct f-Shmanipulation as a reduction to f-Manipulation with a polynomial number of oracle calls.

   Let us say that we have a set of m total alternatives. Then, for each alternative a_i from $i = 0$ to m such that $a_i \succ f(\boldsymbol{\sigma})$, we can define the f-Manipulation as follows:

   - The votes of the non-manipulators are $\boldsymbol{\sigma}_{-i}$.
   - The preferred alternative is a_i
   - The voting rule is f.

   When we run the polynomial-time algorithm for f-Manipulation, it will determine whether there is a vote that σ_i' can cast such that $a_i = f(\sigma_i', \boldsymbol{\sigma}_{-i})$. If we find a a_i where this can occur, then we have found a solution to the f-Shmanipulation and can stop the algorithm. If we do not find a a_i where this can occur, then there is no possible ranking.

   Let us say that the runtime of f-Manipulation is $O(n^p)$. The runtime of our overall algorithm in the worst-case is $m * O(n^p)$, which is also in polynomial time. Thus, if the voting rule f is such that there is a polynomial-time algorithm for f-Manipulation, then there is a polynomial-time algorithm for f-Shmanipulation.

Name: ____________________________________

Additional Space / Scratch Paper

Name: ______________________________

Additional Space / Scratch Paper