CS 580: Algorithmic Game Theory, Fall 2026
HW 3 (due on Friday, 10th November at 11:59pm CST)

Instructions:

1. We will grade this assignment out of a total of 30 points.
2. You can work on any homework in groups of ($\leq$) two. Submit only one assignment per group. First submit your solutions on Gradescope, and you can add your group member after submission.
3. If you discuss a problem with another group then write the names of the other group's members at the beginning of the answer for that problem.
4. Please type your solutions if possible in Latex or doc whichever is suitable, and submit on Gradescope.
5. Even if you are not able to solve a problem completely, do submit whatever you have. Partial proofs, high-level ideas, examples, and so on.
6. Except where otherwise noted, you may refer to lecture slides/notes. You cannot refer to textbooks, handouts, or research papers that have not been listed. If you do use any approved sources, make sure you **cite them appropriately**, and make sure to **write in your own words**.
7. No late assignments will be accepted.
8. By AGT book we mean the following book: Algorithmic Game Theory (edited) by Nisan, Roughgarden, Tardos and Vazirani. Its free online version is available at Prof. Vijay V. Vazirani's webpage.

---

1. (*Recommendation: Do not use LLMs*)
   (a) Consider the following allocation rules for a single-parameter environment auction. For each of these, prove if they are implementable or not.
      i. (2.5 points) The allocation rule that maximizes the social welfare among all feasible allocations. That is, for bids $b_1, b_2 \ldots, b_n$, $x(b) = \text{argmax}_x \sum_i b_i x_i$.
      ii. (2.5 points) Suppose the set of feasible allocations is $X = \{x \in [0,1]^n \mid \sum_i x_i = 1\}$. Give each agent a fraction of the item proportional to the distance of their bid from the average of all bids (normalized to 1), and additionally have $x_i = 0$ when $b_i = 0$. That is, if $b_1, b_2, \ldots, b_n$ are the bids of all agents, and if we let $b_{avg} = \sum_i b_i/n$, then the allocation $x(b)$ satisfies for all agents i, j where $x_i, x_j > 0$, $\frac{x_i}{x_j} = \frac{|b_i - b_{avg}|}{|b_j - b_{avg}|}$, and $\sum_i x_i = 1$.

(b) (5 points) Recall the sponsored search (keyword) auction: k ad slots are on sale on a search website, where probability of getting i^{th} slot clicked is α_i. We know that $\alpha_1 > \alpha_2 > \cdots > \alpha_k > 0$. Given bids $b_1 \geq b_2 \geq \ldots b_n$ of n agents, agent i gets slot i for $i \leq k$, and the remaining agents do not get anything. Note that this allocation rule is monotone. Compute the Myerson's payment for this sponsored search auction.

2. (*Recommendation: Feel free to use LLM/AI tools. But write the answer in your own language making sure you understand the solution. Cite your sources.*)

   Consider an atomic selfish routing game in which all players have the same source vertex and sink vertex (and each controls one unit of flow). Assume that edge cost functions are non-decreasing, but do not assume that they are affine. Prove that a pure-strategy Nash equilibrium can be computed in polynomial time. Be sure to discuss the issue of fractional vs. integral flows, and explain how (or if) you use the hypothesis that edge cost functions are non-decreasing.

   [Hint: Recall the Rosenthal's potential function.]

3. (**Experimental.**) Run the following experiment using any freely available LLM models, such as ChatGPT/Gemini/Calude etc.

   (a) Create two agents, and make them compete in single-item auctions under the following formats: (i) Sealed-bid first price, (ii) Sealed-bid second price, and (iii) Ascending/English auction.

   (b) Create two agents, and ask them to bid in SAA for two items A and B. Set the values of the agents as follows:

$$v_1(A) = v_1(B) = 0, v_1(\{A, B\}) = 100. \quad v_2(A) = v_2(B) = v_2(AB) = 75$$

---

4. (*Bonus Problem. Feel free to use LLMs*) Consider n identical machines and m selfish jobs (the players). Each job j has a processing time p_j. Once jobs have chosen machines, the jobs on each machine are processed serially from shortest to longest. (You can assume that the p_j's are distinct.) For example, if jobs with processing times $1, 3$, and 5 are scheduled on a common machine, then they will complete at times $1, 4$, and 9, respectively. The following questions concern the game in which players choose machines in order to minimize their completion times. The objective function as a planner is to minimize the total completion time $\sum_{j=1}^{m} C_j$, where C_j is the completion time job j.

   (a) Define the rank R_j of job j in a schedule as the number of jobs on j's machine with processing time at least p_j (including j itself). For example, if jobs with processing times $1, 3$, and 5 are scheduled on a common machine, then they have ranks $3, 2$, and 1, respectively.

   Prove that in these scheduling games, the objective function value of an outcome can also be written as $\sum_{j=1}^{m} p_j R_j$.

(b) Prove that the following algorithm produces an optimal outcome: (i) sort the jobs from largest to smallest; (ii) for $i = 1, 2, \ldots, m$, assign the i^{th} job in this ordering to machine i mod n (where machine 0 means machine n).

(c) Prove that for every such scheduling game, the expected objective function value of every coarse correlated equilibrium is at most twice that of an optimal outcome.