50.050/50.550 – Advanced Algorithms

January–April Term, 2026

Homework Set 3

Due by: Week 5 Friday (27 February 2026) 1pm.
Please submit your homework online via eDimension.

NOTE: There are bonus marks (open to both undergraduate students and post-graduate students) at the end of this homework set!

Question 1. Draw all possible isomorphism classes of simple undirected graphs with exactly 5 vertices. How many such isomorphism classes are there? Which of these are isomorphism classes of planar graphs? To get full credit for this question, organize your drawings based on at least one graph invariant of your choice that is not " number of vertices". [5 marks]
Note: Please be neat with your graph drawings!

Question 2. Let $G = (V, E)$ be a triangle-free simple undirected graph, such that no edge-induced sugraph of G is isomorphic to C_4. Assume that V is non-empty, and suppose every vertex $v \in V$ has degree $\deg(v) \geq 10$. Prove that G has at least 101 vertices. [5 marks]
Hint: Recall that a graph G is called *triangle-free* if none of the vertex-induced subgraphs of G are isomorphic to K_3; see also L03.01 slide 18 and L04.03 slide 9.

Question 3. Consider a 12-by-12 checkerboard, which has a total of 144 squares. We say that two squares are *neighboring* squares if they share a common edge. This means that each of the four corner squares of the checkboard has 2 neighboring squares, each of the 40 non-corner squares touching the perimeter of the checkboard has 3 neighboring squares, and each of the remaining 100 "interior" squares of the checkerboard has 4 neighboring squares.

Suppose that we fill all 144 squares of the checkerboard with the integers $1, \ldots, 144$, so that no two distinct squares are assigned the same integer. Prove that we can find two neighboring squares, such that for their assigned integers, their absolute difference is at least 7. [5 marks]

Tip: It would be good to set up the necessary notation and introduce useful terminology. Having clear notation/terminology is the first step in writing a clear proof.

Question 4. Let $G = (V, E, \phi)$ be an undirected bipartite graph with bipartition (V_1, V_2). Suppose that $|N_G(U)| \geq |U|$ for all subsets $U \subseteq V_1$. In Week 4's cohort class, we covered the intuition for an explicit construction of a complete matching M from V_1 to V_2, as guaranteed by Hall's marriage theorem. En route to this explicit construction, a key step was the modification of a non-maximum matching M in G into a new matching M' in G, such that $|M'| = |M| + 1$. Starting with M, we constructed a sequence of vertices $(x_0, y_1, x_1, y_2, x_2, \ldots, y_\ell, x_\ell, y_{\ell+1})$, such that x_i is a vertex in V_1 for all $0 \leq i \leq \ell$, and y_i is a vertex in V_2 for all $1 \leq i \leq \ell + 1$, as detailed in the cohort class slides ("Intuition for Hall's marriage theorem"). In your own words, explain why it is not always true that there are $\ell + 1$ edges $e_0, e_1, \ldots, e_\ell$ in E such that $\phi(e_i) = \{x_i, y_{i+1}\}$ for all $0 \leq i \leq \ell$. (Hint: Providing an example would be useful). Please give as much details as possible. We are specifically looking out for evidence of your understanding on why the described matching algorithm works as intended. [5 marks]

Question 5. (**For post-graduate students only**) In Week 4's cohort class, we learned an explict algorithm for computing complete matchings, which serves as an introduction to matching algorithms. Hall's marriage theorem and complete matchings are essentially the starting point of what is called "matching theory", which can be thought of as a vast extension of the notion of "matching" that we covered in class, beyond the context of bipartite graphs, and even beyond unweighted graphs, where we incorporate constraints, preferences, and utility functions.

In your research career, when you write research papers, novelty is always of utmost importance in getting your research published. Although it helps to read up on existing papers that are closely related to your intended research problem(s), what would make your paper stand out is to incorporate ideas that are unexpected, new, and which are NOT found in existing closely related papers. Hence, it is very useful to learn and be aware of diverse mathematical ideas, some of which may initially not even seem related to your research. You would never know when you get inspiration and establish new connections, between your specific research area, and some new idea you have learned (whether from class or from reading a paper that is not obviously closely related).

For this HW question, we shall simulate the process of getting inspiration. Read the following survey paper on matching algorithms:

- Paper title: "Matching Algorithms: Fundamentals, Applications and Challenges"
- Authors: Jing Ren, Feng Xia, Xiangtai Chen, Jiaying Liu, Mingliang Hou, Ahsan Shehzad, Nargiz Sultanova, Xiangjie Kong.
- Journal: IEEE Transactions on Emerging Topics in Computational Intelligence
- ArXiV link: `https://arxiv.org/abs/2103.03770`

After reading this paper, propose a concrete research problem that may be of interest to you, and of relevance to your research area, that incorporates matching algorithms. When formulating your research problem, you should be explicit about the exact type of matching to be used, and you should try to be as precise as possible in your mathematical set-up of your problem formulation. You should also include at least a brief introduction of the context of your research problem, so that it is understandable to those not directly in your research area.

[5 marks, for clarity]

Bonus Question.[1] (**open to both undergraduate and post-graduate students**)
John, our friendly blue bird, has a treasure vault at home. To protect his treasure inside the vault, John has installed a very high-tech security system, which he claims is very very secure, because it has what he calls a "thirty-factor authentication" system. Essentially, to unlock the vault, a total of 30 different passwords must be keyed into this security system. Unfortunately, John easily forgets passwords, so he has entrusted 7 of his friends with the passwords, such that the following conditions are satisfied:

- Every friend (among these 7 friends) knows at least one password.
- Every possible group of 4 friends among these 7 friends would collectively know all 30 passwords. (There are $\binom{7}{4}$ such groups.)

Prove that we can find some 3 friends (among these 7 friends) who would collectively know all 30 passwords. [7 bonus marks]

[1] This bonus question is purely optional. If you solve this completely correctly, you get 7 bonus marks. Partial bonus marks are awarded for partially correct solutions. In this course, bonus marks will count towards the 20% homework component of your final grade. As this question is purely optional, no suggested solution will be provided.