CSE 521: Design and Analysis of Algorithms	Fall 2025

Problem Set 6 (Midterm)

Deadline: Nov 13th in gradescope

You cannot collaborate on this HW!

P1) Recall a set of vectors $v_1, \dots, v_n \in \mathbb{R}^d$ are linearly independent if for any set of coefficients $c_1, \dots, c_n \in \mathbb{R}$, $c_1 v_1 + \dots + c_n v_n \neq 0$. Further, as discussed in class, $\text{rank}(A)$ of a matrix A is the maximum number of linearly independent columns of A and the maximum number of linearly independent rows, and the number of non-zero singular values of A. Also, recall that for a matrix A, $\det(A) \neq 0$ if and only if $\text{rank}(A) = n$. It can be shown that for any matrix A, $\det(A) \neq 0$ if and only if A is nonsingular.

Let $G = (X, Y, E)$ be a given bipartite graph with $|X| = |Y| = n$. Using the Schwartz-Zippel lemma, we can rewrite the algorithm that tests whether G has a perfect matching as follows: For each edge x_i, y_j of G, choose $A_{i,j}$ uniformly and independently from the set $\{0, 1, \dots, n^2\}$, and let the rest of entries of A be 0. Return yes if $\text{rank}(A) = n$ and no otherwise.

a) Let A be the following matrix: For each nonadjacent pair x_i, y_j, let $A_{i,j} = 0$; choose the rest of the entries of A arbitrarily (instead of random). Use properties of the rank and determinant to show that if $\text{rank}(A) = k$, then G has a matching of size at least k.

b) Use the Schwartz-Zippel lemma to design a randomized algorithm to compute the size of the maximum matching of G. Can you upper bound probability of failure of your algorithm? In your algorithm assume that you can compute the rank of a matrix in polynomial time.

P2) In this problem we prove that for any (undirected unweighted) graph G, $\chi(G) \leq \lambda_1(A_G) + 1$, that is we can color vertices of G with (at most) $\lambda_1(A_G) + 1$ colors such that every adjacent pair of vertices have distinct colors where $\lambda_1(A_G)$ is the largest eigenvalue of the adjacency matrix of A. For example if G is a triangle,

$$A_G = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

Then, $\lambda_1(G) = 2$ and we can color vertices of G with 3 colors, 1 with A, 2 with B and 3 with C.

a) Show that $\lambda_1(A_G)$ is at least the average degree of G, i.e., $\lambda_1(A_G) \geq \text{avg}_i d_i = \frac{1}{n} \sum_i d_i$.

b) For a set $S \subseteq V$ let $G[S]$ denote the induced subgraph of G on S, i.e., the graph with vertex set S where any pair is connected iff they are connected in G. Show that for any $S \subseteq V$, $\lambda_1(A_G)$ is at least the average degree of vertices of $G[S]$.

c) Design an algorithm that colors the vertices of G with at most $\lambda_1(A_G) + 1$ many colors.