Exercise Set XII, Advanced Algorithms 2022

These exercises are for your own benefit. Feel free to collaborate and share your answers with other students. Solve as many problems as you can and ask for help if you get stuck for too long. Problems marked * are more difficult but also more fun :).

These problems are taken from various sources at EPFL and on the Internet, too numerous to cite individually. **The problems are not ordered with respect to difficulty.**

1 Let $G = (V, E)$ be a d-regular undirected graph, and let $M = \frac{1}{d}A$ denote its normalized adjacency matrix. Let $1 = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ denote the eigenvalues of M. Prove that if $\lambda_{n-1} = -1$, then G is disconnected.

2 Let $G = (V, E)$ be a d-regular undirected graph, and let $M = \frac{1}{d}A$ denote its normalized adjacency matrix. Let $1 = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ denote the eigenvalues of M. Prove that if $\lambda_n > -1/2$, then G is not tripartite (recall that a graph is tripartite if its vertex set can be partitioned into three disjoint sets such that all of its edges connect vertices in different components).

3 Consider a family $\mathcal{H}$ of hash functions $h : [n] \to [n]$. We want to analyze the quantity

$$\text{maxload(h)} := \max_{b \in [n]} |\{h(i) = b\}|$$

when $n \geq 10$.

a Assuming $\mathcal{H}$ is a pairwise independent family of hash function, prove that

$$\Pr_{h \in \mathcal{H}}[\text{maxload(h)} > 11\sqrt{n}] \leq 1/100\,.$$

b Assuming that $\mathcal{H}$ is the family of completely random hash functions, prove that

$$\Pr_{h \in \mathcal{H}}[\text{maxload(h)} > 9\ln(n)] \leq 1/2\,.$$

4 Consider an undirected graph $G = (V, E)$. Show that the following linear program, that has a variable x_v for each $v \in V$, can be solved in polynomial time

$$\begin{array}{lll}
\textbf{maximize} & \displaystyle\sum_{v \in V} x_v & \\
\textbf{subject to} & \displaystyle\sum_{v \in S} x_v \leq |\delta(S)| & \text{for every } S \subsetneq V \\
& \quad x_v \geq 0 &
\end{array}$$

Recall that $\delta(S)$ is the set of edges crossing the cut S.

By Ellipsoid method this reduces to the following separation problem: Given x, design a polytime algorithm that verifies whether x is feasible or, if not, outputs a violated constraint.