15-451 Spring 2026 Problem List

Instructors and TAs, Carnegie Mellon University

March 30, 2026

Problem list is organized by units, which correspond to content on exams. All problems in this list are covered either in lecture, (oral) homework, or recitation. All exam problems will come from the problem list, or are a simplified version of a problem on the list. It is possible that problems will be deferred to later exams, depending on how much progress we make in lectures.

Problems are marked by difficulty from one star to three stars, and their source (lecture, HW, etc.).

Contents

1 Midterm 2: Amortized Analysis and Combinatorial Optimization

This section contains problems that may appear on **Midterm 2**.

1. (* HW 3 p1, with DP removed) Given an array $a[1 \dots n]$ of length n, as well as two monotone sequences of left and right indices $1 \le l[1] \le l[2] \le \dots \le l[n]$, $1 \le r[1] \le r[2] \le \dots \le r[n]$, such that $l[i] \le r[i]$. Compute, for each i, the range minimum/maximum in $a[l[i] \dots r[i]]$ in $O(n)$ time.

2. (** HW 3, upgraded) In an undirected graph with colors on the n vertices undergoing m edge insertions, track the number of pairs of vertices of the same color that can reach each other in time $O((m + n) \log^2 n)$ (or better). Note that there may be up to n colors: Quiz 3/HW 3 asked for the version with 2 colors.

3. (** HW4, with story involving foxes removed) Given sets A, B, C, each of size n, as well as some admissible pairs $E \subseteq A \times B$ and $F \subseteq B \times C$, compute, using a single maximum-flow call, the maximum number of pairwise disjoint triples a_i, b_i, c_i such that $(a_i, b_i) \in E$ and $(b_i, c_i) \in F$ for all i.

4. (***, Recitation 7) Give an $O(m \log n)$-time algorithm to decide whether an undirected graph with edges weighted either 0 or 1 has a spanning tree with weight exactly k. Justify the correctness of this algorithm.

5. (* HW5) Let G be a directed graph with m edges whose edges have positive lengths. Let s be a source vertex, and let $v_1, v_2, \dots, v_k$ be other vertices given as input. Your goal is to return edge-disjoint paths $P_1, P_2, \dots, P_k$, where P_i is from s to v_i, and minimize the total sum of their lengths. Formulate this as a linear program in standard form.

6. (* Oral HW3) Consider the following flow-based view of Markov Decision Processes:

   (a) Flows still need to be in the direction of the edge, and must respect capacities.

   (b) There are now two types of vertices. The ones from max-flow / min-cost flow are now "conservation vertices," where the incoming flow can be distributed arbitrarily among the outgoing edges.

   (c) There are also "random walk" vertices, which have out-degree 2: any flow that arrives at such a vertex gets distributed evenly among the two outgoing edges.

   Again, we want to route one unit of flow from s to t, satisfying conservation at all other vertices, while minimizing the total cost on the edges, subject to capacity constraints on edges.

   Formulate this as a linear program in standard form.

7. (*, Recitation 8) Show that in a directed graph with capacity *cap*, the fractional s-t minimum-cut linear program can be written as

$$\min_{\substack{x \in \mathbb{R}^V, x_s = 0, x_t = 1 \\ y \in \mathbb{R}^E, y \geq 0 \\ y \geq Bx}} \text{cap}^\top y \tag{1}$$

where recall $B \in \mathbb{R}^{m \times n}$ is the edge-vertex incidence matrix, with an edge $e = (u \to v)$ corresponding to a row with $B_{e,u} = -1$, $B_{e,v} = 1$, and 0 everywhere else.

8. (**, HW5) Let G be a directed graph, and let $(s_1, t_1), \ldots, (s_k, t_k)$ be pairs of sources and sinks. We want to send one unit of flow for each pair $s_i \to t_i$ (**Important: the flows need not be integral.**) in a way that minimizes the maximum congestion on any edge. Formally, let $f^{(1)}, \ldots, f^{(k)}$ be the flows. Then the congestion of edge e is defined as

$$\text{cong}_e = \sum_{i=1}^{k} f_e^{(i)},$$

where $f_e^{(i)}$ denotes the amount of flow on edge e in $f^{(i)}$. You want to find flows $f^{(i)}$, each routing one unit from s_i to t_i, that minimize $\max_e \text{cong}_e$.

9. (** Oral HW 3) Consider the following game played on the edges of a directed graph with two special vertices s (source) and t (sink). The following actions are taken simultaneously (using randomized strategies):

   (a) A picks a path from s to t.

   (b) B picks an edge.

   B wins if the edge picked is on the path; otherwise, A wins.

   Show that this is a zero-sum game, formulate this as a linear program, and relate the objective of this game to the s-t max-flow/min-cut in this graph.

10. (***, HW5) Let G be a bipartite graph with bipartition $V = L \cup R$, where L is the set of left vertices and R is the set of right vertices. Let $a = |L|$ and $b = |R|$ (L has a vertices and R has b vertices). Define a 2-matching of G to be a subset of edges $F \subseteq E(G)$ such that:

    (a) Every vertex in L is incident to at most 2 edges of F, and

    (b) every vertex in R is incident to at most one edge of F.

    Prove that the maximum number of edges in a 2-matching is given by the following formula:

$$2a - \max_{S \subseteq L} \left(2|S| - |N(S)|\right),$$

    where $N(S) \subseteq R$ is the set of neighbors in R of the vertices in S.

11. (*, Lecture 17, Recitation 9) Given some explicit 2×2 or (simple) 3×3 payoff matrix, be able to:

    - Write down a linear program whose value is the optimal game value (for the versions where player 1 reveals their strategy and where player 2 reveals their strategy), and
    - solve for the optimal game values and a pair of equilibrium strategies.