Homework 7:

Due: November 4, 2025 at 2:30p.m.

This homework must be typed in LaTeX and submitted via Gradescope.

Please ensure that your solutions are complete, concise, and communicated clearly. Use full sentences and plan your presentation before your write. Except where indicated, consider every problem as asking for a proof.

Problem 1. Given a flow network graph G with n nodes and m edges, assume you are given a maximum flow assigment. Propose an algorithm that finds a minimum capacity cut in time $O(m)$.

Problem 2. Karp-Rabin patern matching algorithm is particularly useful in multidimensional pattern matching.
Let T be an array of $n \times m$ characters, and let P be a pattern of $p \times q$ characters, $p << n$ and $q << m$.
We hash a pattern of $p \times q$ characters as follows: We first hash each row $1 \leq i \leq p$ (which is a string of q characters) to a value $h(i)$. We then hash the string of p values $(h(1), \ldots, h(p))$ to one value $h_2(h(1), \ldots, h(p))$.
We use a "rolling hash" function for both the row and column hashing.
To search for pattern P in T we compute the h_2 value of each $p \times q$ block by applying the rolling hash function h to each row, and then the rolling hash function h_2 to compute the h_2 value of the block.

- Write a (pseudo code) program that implements this search with no more than $O(p+q)$ extra memory.
- Show that the time complexity of your program is $O(nm + pq)$. (You can ignore the cost of verifying patterns found by the hashing process.)

Problem 3.
Let T be a set of teams in a sports league, which, for historical reasons, we assume is baseball. At any point during the season, each team, i, in T, will have some number, w_i, of wins, and will have some number, g_i, of games left to play. The baseball elimination problem is to determine whether it is possible for team k to finish the season first, given the games it has already won and the games it has left to play. Note that this depends on more than just the number of games left for team k, however; it also depends on the respective schedules of team k and the other teams. (See example in Table 1.)

We can solve this problem by reduction to a network flow problem:

Let $g_{i,j}$ denote the number of games remaining between team i and team j.
Let $g_i = \sum_{j \in T \setminus \{k\}} g_{i,j}$.
w_i is the current number of wins for team i.
Let $T' = T \setminus \{k\}$ be the rival teams of k.
Let L be the set of matches left to play between teams in T'.

$$L = \{\{i, j\} : i, j \in T' \text{ and } g_{i,j} > 0\}$$

Let

$$W = w_k + g_k$$

be the aximum number of possible wins for k, and assume

$$W \geq \max_i w_i.$$

To consider how a combination of teams and game outcomes might eliminate team k, we create a graph G:

Vertices: $\{s, t\} \cup L \cup T'$

Edges:

- For each $\{i, j\} \in L$, add edge $(s, \{i, j\})$ with capacity $g_{i,j}$.
- For each $\{i, j\} \in L$, add edges $(\{i, j\}, i)$ and $(\{i, j\}, j)$ with capacity ∞.
- For each team i, add edge (i, t) with weight $W - w_i > 0$.

1. Let f be the maximum flow in this graph. Prove that there exists a combination of results for which team k wins the championship if and only if $f = \sum_{i,j \neq k} g_{i,j}$.
2. What is the time complexity of running the Ford Fulkerson flow algorithm on this graph?

Team i	**Wins** w_i	**Games Left** g_i	**Schedule** $(g_{i,j})$			
			LA	Oak	Sea	Tex
Los Angeles	81	8	–	1	6	1
Oakland	77	4	1	–	0	3
Seattle	76	7	6	0	–	1
Texas	74	5	1	3	1	–

Table 1: A set of teams, their standings, and their remaining schedule. Clearly, Texas is eliminated from finishing in first place, since it can win at most 79 games. In addition, even though it is currently in second place, Oakland is also eliminated, because it can win at most 81 games, but in the remaining games between LA and Seattle, either LA wins at least 1 game and finishes with at least 82 wins or Seattle wins 6 games and finishes with at least 82 wins.