CS 161 (Stanford, Fall 2025) Section 8

1 Warm up - True or False

Decide whether you think the following statement is true or false. If it is true, give a short explanation. If it is false, give a counterexample.

Let G be an arbitrary flow network, with a source s, a sink t, and a positive integer capacity c_e on every edge e, and let (A, B) be a minimum s-t cut with respect to these capacities $\{c_e : e \in E\}$. Now suppose we add 1 to every capacity; then (A, B) is still a minimum s-t cut with respect to these new capacities $\{1 + c_e : e \in E\}$.

2 Max Flow Potpourri

How would you use a max flow algorithm to handle the following situations?

1. Suppose that instead of having a single source s and a single sink t, we have multiple sources $S = \{s_1, s_2, ..., s_k\}$ and multiple sinks $T = \{t_1, t_2, ..., t_\ell\}$. How can you find the max flow in the graph from sources to sinks?
2. Suppose that in addition to edges having max flow capacities, vertices also have a limit to their capacity; that is, each vertex v_i has capacity c_i. How can you find the max flow from a source s to sink t in this graph?

3 Task Selection

Suppose you have a set of k tasks $t_1, ..., t_k$. There are certain tasks such that t_i is a prerequisite of t_j. Each task t_i also has an integer reward r_i, which may be negative. Find an optimal subset of tasks to complete to maximize your reward.

4 Tiling Partial Checkerboard

Suppose you are given an $n \times n$ checkerboard with some of the squares deleted. You have a large set of dominoes, each of which is just the right size to cover two adjacent squares of the checkerboard. Describe and analyze an algorithm to determine whether one can completely tile the board with dominoes. Each domino must cover exactly two adjacent undeleted squares, and each undeleted square must be covered by exactly one domino.

Your input is a two-dimensional $n \times n$ array $Deleted$, where $Deleted[i, j]$=True if and only if the square in row i and column j has been deleted. Your output is a single bit; you do not have to compute the actual placement of dominoes.