CS 161 Fall '18 Section 6

1 Warm up - True or False

Decide whether you think the following statement is true or false.

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 mimimum $s - t$ 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 Potpurri

(a) Suppose that instead of having a single source and sink s, t respectively, we have multiple sources $S = \{s_1, s_2, ..., s_k\}$ and multiple sinks $T = \{t_1, t_2, ..., t_\ell\}$. We wish to still find the max flow in the graph from sources to sinks.

(b) 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. We wish to find the max flow from a source s to sink t in this graph.

(c) Given a solution to max-flow, verify that it is correct in linear time.

3 Minesweeper

This problem develops some of probabilistic intuition useful for this week's Thursday lecture (L16).

Suppose that you're playing a version of minesweeper where there are n land slots and only one of them contains a mine. When you visit a land slot, you will explode if the slot contains the mine. If the slot contains no mine, no additional information is revealed about any other slot (i.e. there are no numbers telling you how many adjacent slots contain mines, etc.). If you explode, you start over with a new board of size n and worst-case location of the mine. You decide to use a randomized algorithm—choosing uniformly at random from the remaining slots—to make your decisions about which slot to visit next. The game ends when you visit all $n - 1$ safe slots on a single board without exploding.

(a) What is the probability of succeeding in one run without ever exploding?

(b) What is the expected number of slots visited until you win?

(c) How does your answer to (b) change if, after you explode, you are allowed to restart using the randomized algorithm at the point right before you exploded (e.g. if you explode with 10 slots (9 safe and 1 mine) left, your next move will be chosen uniformly at random from the same 10 slots)?

(d) What if you only find out at the end of the game whether you exploded or succeeded, but if you exploded you can restart from any point in history?

4 Expense Settling

You've gone on a trip with k friends, where friend i paid c_i for the group's expenses. You would like to develop an algorithm to ensure that everyone gets paid back fairly, but without going through one person (that is, each person would either pay or receive money, but not both).

5 Project 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 also has a reward r_i, which may be negative. Find an optimal subset of tasks to complete to maximize your reward.

6 Tiling Partial Checkerboard

Suppose you are given an $n \times n$ checkerboard with some of the squares deleted. You have a large set of dominos which are each 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 dominos. 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 dominos.