

Homework 6

Due Date: November 14, 2019 at 11:59PM on Gradescope.

Instructor: Jeremiah Blocki

Homework Guideline Reminders

- Assignments must be typed. Submit one pdf file to Gradescope by 11:59PM, or else late penalties will apply. The pdf file can include hand-drawn images of figures.
- Each question needs to start with the resources and collaborator (RC) statement. You will not be penalized for using resources or having collaborators if your answers are expressed *entirely* in your own words. If you consulted no resources outside of course material or had no collaborators, you must state so. A question without a complete RC statement will not be graded.

Question 1: Where Are We Eating?

Jane wants to have a dinner party with her $n - 1$ friends, but they simply can't agree on a place to eat. Each person i (including Jane) has a preferred restaurant $R[i]$ with a unique price $C[i]$. Let $G = (V, E)$ be a directed “influence” graph with n nodes and m edges. Each node in G represents one of Jane or her friends, and a path from node i to j represents “influence” from person i to person j . When casting his or her vote, person j will consider voting for restaurant $R[i]$ if and only if person j is influenced by person i (note that person j is influenced by his/her own preference) — person j will vote for the cheapest restaurant under consideration.

At the final vote, every person will vote for the restaurant with the least price out of all of their willing restaurants. Consider the following example:

Person	R	C
Jane (J)	Dynamic Pizzeria	8
Elizabeth (E)	Gushing Gummy Bears	20
Lydia (L)	Greedy Grill	16
Charles (C)	Dinner & Coffee	14
William (W)	Big O Bar	12
George (G)	NP Cafe	10

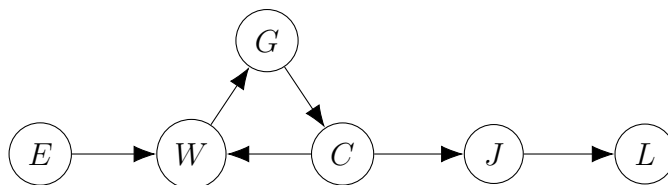


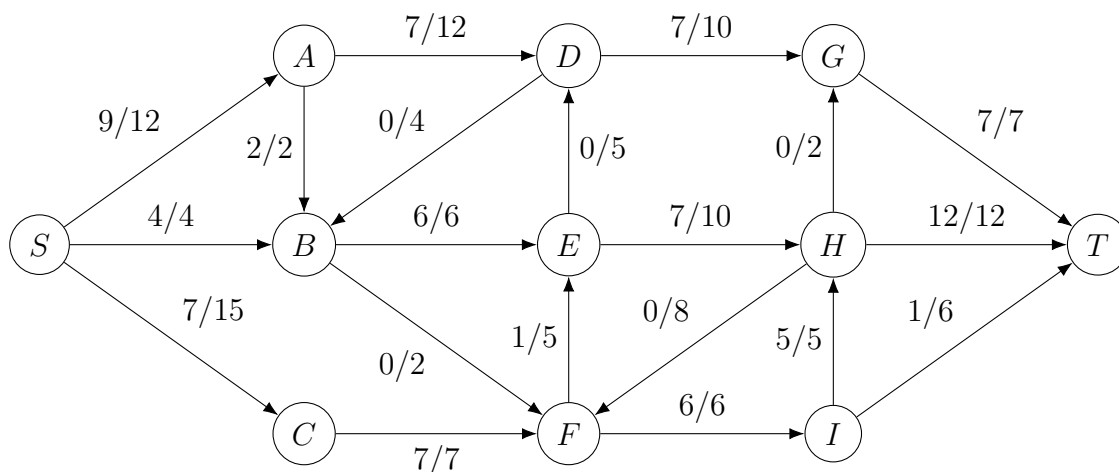
Figure 1: Influence Graph G

In this example, Elizabeth is not influenced by anybody, so she only considers voting for Gushing Gummy Bears. William, George, and Charles are all influenced by Elizabeth and each other, so they are each willing to consider four restaurants: Gushing Gummy Bears, Dinner & Coffee, Big O Bar and NP Cafe. Since NP Cafe is the cheapest option of these four options William, George and Charles will all vote for NP Cafe. Jane is influenced by everybody else except Lydia, so she considers voting for every restaurant except Greedy Grill. Jane votes for her preference of Dynamic Pizzeria due to the price. Lydia is influenced by everybody and considers voting for all restaurants. She votes for Dynamic Pizzeria — the cheapest restaurant. Hence, it can be determined that Gushing Gummy Bears will have 1 vote, NP Cafe will have 3 votes, Dynamic Pizzeria will have 2 votes, and all others have no votes, so the entire group will eat at NP Cafe.

Help Jane and her friends choose a place to eat by designing an efficient algorithm that determines the restaurant with the most votes (breaking ties arbitrarily). Briefly argue the correctness of your algorithm (no formal proof required). Analyze the time complexity of your algorithm. An $O(m + n)$ runtime solution is required for full credit, but sub-optimal solutions will be accepted for partial credit.

Question 2: Max Flow Exercise

A max-flow algorithm has been performed part-way to achieve the flow in the network shown below. Edges represent current capacity out of max capacity. To answer the following questions, you may include photographs of hand-drawn graphs.



(a) Draw the residual graph for this flow network.

- (b) Run one iteration of Ford Fulkerson. Show the augmenting path and draw the flow network with the updated flow.
- (c) What is the partition of vertices for the minimum cut? Explain how you arrived at your answer.
- (d) What is the max-flow of this network?

Question 3: It's Party Time!

Elizabeth is an event planner at Prof. Blocki's event management company. Christmas party season is right around the corner and Prof. Blocki is organizing m parties. Since Prof. Blocki is very popular, Elizabeth has requests from n people who wish to attend these parties.

For each person i you are given the list of parties P_i that i is willing to attend. You are also given the maximum capacity of each party, j , denoted by M_j . Additionally, some of the attendees are married. You are given a list of couples, C , where if $\{a, b\}$ is in C , then person a and person b are married. Of course, marriage is symmetric (i.e. if a is married to b , then b is married to a), and each person is only married to at most one other person. Married attendees must avoid their spouse, otherwise they will get angry at each other's party habits, so a person will not attend a party that their spouse is attending. Each person has time to attend at most 2 parties.

Devise an efficient algorithm to determine the maximum total attendance possible to all parties given the constraints. Briefly argue the correctness of your algorithm (no formal proof required). Analyze the time complexity of your algorithm.

Question 4: Capacity Scaling Proof

Let G be a flow network with integral capacities $c(e) \geq 1$ on each edge e and let $C = \max_e c(e)$ be the maximum capacity of any edge e in the flow network. Let v^* denote the value of the maximum flow. In this question we will prove that the Capacity Scaling algorithm from lecture computes a maximum flow of value v^* in polynomial time in n and $\log C$.

1. Explain why the outer while loop repeats at most $1 + \lceil \log_2 C \rceil$ times.
2. Suppose that the edge $e = (u, v)$ in G has capacity $c(e)$ and let f be any flow. Lower bound $f(e)$ in terms of $c(e)$ and Δ assuming that the edge e does not appear in $G_f(\Delta)$. Upper bound $f(e)$ assuming that the reversed edge (v, u) does not appear in $G_f(\Delta)$.
3. Let f be a flow at the end of a scaling phase (i.e., such that $G_f(\Delta)$ has no augmenting path). Prove that $v^* \leq v(f) + m\Delta$. (**Hint:** It may be useful to think about running a BFS in $G_f(\Delta)$).
4. Prove that each scaling phase ends after *at most* $O(m)$ augmentations. (**Hint:** Try to lower bound the amount each augmentation increases the flow).
5. Upper bound the total running time of the Capacity scaling algorithm.