CS 170 Homework 9

1 Study Group

List the names and SIDs of the members in your study group. If you have no collaborators, explicitly write "none".

2 Mittens

You are running Toasty Digits, a company that produces mittens. To make sure that your company can meet demands, you are planning out the production for the coming n months. The following information is given to you:

- **Demand:** In month i, the demand will be $d_i \geq 0$ (i.e. you sell exactly d_i pairs of mittens in month i). The total demand for all n months is given by $D = \sum_{i=1}^{n} d_i$.
- **Production:** Your full-time knitting staff can produce at most m pairs of mittens per month. You can hire additional knitters who will produce as many additional mittens as you need, but at a cost of c dollars per pair (whereas you do not pay anything per pair for your full-time staff).
- **Storage:** If, at the end of a month, you have any leftover mittens, you have to store them at a cost. In particular, if you have k pairs of mittens left, you pay $h_k \geq 0$ dollars. You can store at most D pairs of mittens.

Provide a dynamic programming algorithm for computing the minimum total cost required to meet all demand. Your solution should include the following:

(a) A description of your subproblems and how you will use them to compute the final answer.

Hint: Let one of the subproblem parameters be the number of leftover mittens.

(b) A recurrence relation for your subproblems and the relevant base cases.

(c) A justification for your recurrence relation and your base cases.

(d) The order in which to solve the subproblems.

(e) The runtime of solving all subproblems and computing the final answer.

3 Simplex Practice

(a) Consider the following linear program:

$$\begin{aligned} \text{Maximize } & x + y \\ \text{subject to: } & -\frac{1}{2}x + y \leq 3 \\ & x + 2y \leq 12 \\ & y \leq 4 \\ & 3x + y \leq 21 \\ & x, y \geq 0 \end{aligned}$$

Draw the feasible region. Write out the sequence of vertices traversed by the simplex algorithm starting at $(0, 3)$.

(b) Consider the following linear program on three variables:

$$\begin{aligned} \text{Maximize } & f(x, y, z) \\ \text{subject to: } & x + 2y \leq 10 \\ & z \leq 3 \\ & x, y, z \geq 0 \end{aligned}$$

Is it possible for $p = \{(0, 0, 0), (0, 5, 0), (10, 0, 0), (10, 0, 3)\}$ to be a valid path that the simplex algorithm takes for some linear function f? If so, is there a linear function, f, such that p is the *only* possible valid path that the simplex algorithm may take?

(c) Over all linear programs in two variables and n constraints (including $x_1, x_2 \geq 0$), determine the minimum and the maximum number of vertices that the simplex algorithm may need to traverse over including its starting and ending points. You may assume that we start already at some vertex on the feasible region, $\vec{x}$.

(d) In the simplex algorithm, we can choose *any* better vertex upon each iteration. What if we make the following modification: choose the *best* adjacent vertex. How does this change the answer to part (c)?

4 Max-Flow Min Cut Basics

For each of the following, state whether the statement is True or False. If true provide a short proof, if false give a counterexample.

(a) If all edge capacities are distinct, the max flow is unique.

(b) If all edge capacities are distinct, the min cut is unique.

(c) If all edge capacities are increased by an additive constant, the min cut remains unchanged.

(d) If all edge capacities are multiplied by a positive integer, the min cut remains unchanged.

(e) In any max flow, there is no directed cycle on which every edge carries positive flow.

(f) There exists a max flow such that there is no directed cycle on which every edge carries positive flow.

5 Flow vs LP

You play a middleman in a market of m suppliers and n purchasers. The i-th supplier can supply up to $s[i]$ products, and the j-th purchaser would like to buy up to $b[j]$ products.

However, due to legislation, supplier i can only sell to a purchaser j if they are situated at most 1000 miles apart. Assume that you're given a list L of all the pairs (i, j) such that supplier i is within 1000 miles of purchaser j. Given m, n, $s[1..m]$, $b[1..n]$, and L as input, your job is to compute the maximum number of products that can be sold. The runtime of your algorithm must be polynomial in m and n.

Show how to solve this problem using a network flow algorithm as a subroutine. Describe the graph and explain why the output from running the network flow algorithm on your graph gives a valid solution to this problem.

6 A Cohort of Secret Agents

A cohort of k secret agents residing in a certain country needs escape routes in case of an emergency. They will be travelling using the railway system which we can think of as a directed graph $G = (V, E)$ with V being the cities and E being the railways. Each secret agent i has a starting point $s_i \in V$, and all s_i's are distinct. Every secret agent needs to reach the consulate of a friendly nation; these consulates are in a known set of cities $T \subseteq V$. In order to move undetected, the secret agents agree that at most c of them should ever pass through any one city. Our goal is to find a set of paths, one for each of the secret agents (or detect that the requirements cannot be met).

Model this problem as a flow network. Specify the vertices, edges, and capacities; show that a maximum flow in your network can be transformed into an optimal solution for the original problem. You do not need to explain how to solve the max-flow instance itself.

7 Office Hour Scheduling

You're the new CS 170 head TA and need to assign TAs to host office hours. Unfortunately for you, the department has bizarre rules about when TAs can hold office hours:

- There are N office hours slots in total. You can assume that all N slots are consecutive.
- There are T TAs in total. The i^{th} TA is only available to hold office hours during the contiguous block from slot a_i to slot b_i. The numbers $\{a_i, b_i\}$ for $i \in \{1 \dots T\}$ are given as input.
- Each office hours slot must have exactly k TAs (no more, and no less).
- Not every TA needs to hold office hours.
- If a TA holds office hours, department rules require them to hold office hours for their entire availability, e.g. the i^{th} TA either holds office hours for every slot from a_i to b_i, or does not hold office hours at all.

Your goal is to determine which TAs to assign office hours so that all the above constraints are met.

Devise an efficient algorithm that solves the problem by constructing a directed graph G with a source s and sink t, and computing the maximum flow from s to t.

(a) Describe the nodes of the graph G. How many nodes does G have?

(b) Describe the edges of the graph G (draw a picture if needed). How many edges does G have?

(c) Given a maximum flow in the graph G, how can you determine which TAs to assign to hold office hours?