Homework 5: Not that kind of programming

Version 2 We updated problems 2b and 3 (updates marked in each problem)

Due Date: This assignment is due at 11:59 PM Friday February 19 (Seattle time, i.e. GMT-8).
except for the programming problem, which will be due one week later with HW6.

You will submit the written problems as a PDF to gradescope. Please put each numbered problem on its own page of the pdf (this will make selecting pages easier when you submit).

Remember to assign pages in your submission!

Collaboration: Please read the full collaboration policy. If you work with others (and you should!), you must still write up your solution independently and name all of your collaborators somewhere on your assignment.

Style and Assumptions: You should check the Style Guide for general principles on how to format your answers. You may use any algorithm described in this list without further explanation. If you wish to use an algorithm you learned in 373 (or 142/143) that isn't on the list, please ask on Ed so we can add it to the list for everyone.

1. Finish HW4

Remember to finish the programming question from HW4!

2. Much Needed Rest

You have a tree that represents the supervisor relationships in a company. Every vertex represents a person and every parent-child relationship means the parent is the supervisor of the child node. Your graph might not be binary (i.e. a supervisor might have varying numbers of people directly below them).

It has been decided that the entire company needs to take a week off. But that shouldn't happen all at once. Everyone will take either this week, next week, or the week after off.

But, you need to ensure that no supervisor takes off the same week as one of their direct reports (i.e., no parent and child in the tree can have the same week off). Subject to that condition, your goal is to have as few people as possible take this week off.

More formally, given a tree T, your goal is to assign exactly one of the labels $0, 1, 2$ to every vertex, so that no edge connects identical labels and the minimum number of '0' labels are used.

(a) Give pseudocode for an algorithm to run for this problem (Hint: You don't need DP here! We only needed one line). Give a 1-2 sentence explanation for why your algorithm is correct. [4 points]

(b) Now, suppose that some of the leaves come pre-labeled (some interns have already gotten their time off approved weeks ago). We'll find a dynamic programming algorithm to solve the problem.

- Write a recurrence (or multiple recurrences) that describe the minimum number of 0 labels in a tree where you don't get to change any of the pre-labeled nodes.
  In your recurrence, you may use "if prelabeled as 0" (or similar conditions) as needed.
  If there is no way to assign weeks off (say one supervisor has children pre-labeled with all three possible labels), return $-\infty$. [8 points]
  Update: *You may use ∞ or $-\infty$ (or both) in the case that there is no labeling. Use whichever makes it easier to write the recurrence*
- Give an English description of what each of your recurrence(s) calculate and what each of the parameters means (1-2 sentences each should be enough). [3 points]
- State what inputs to give to (one of your) recurrences that will be the minimum number of people labeled as 0. [1 point]

(c) We said in class that a post-order traversal is usually sufficient to calculate DP on trees. Is that sufficient for your DP? Justify in 2-3 sentences. If your answer is "no" also include 1-2 sentences on what evaluation order you would use instead. [3 points]

(d) What is the running time to evaluate your recurrence? Let n be the number of vertices in your tree. No justification required. [1 point]

3. Linear Photography

You've got 1000 photos that you need labeled (to generate a test set for your ML system). There are two companies (like Mechanical Turk) that will label your photos for you.

Company A keeps things consistent.

- They will charge you 5 cents per photo.
- They have a hard cap of 400 photos (no matter how much you pay, they won't label photo number 401.)

Company B prefers to deal with many customers with small accounts, so charges more for bigger orders

- They will charge 3 cents per photo for your first 200 photos.
- They will charge 4 cents per photo for your next 700 photos.
- They will charge 7 cents per photo for any photo after that.

In this problem you will design linear program(s) to find how many photos to give to each company to pay the least price possible. You do **not** need to solve the LPs and find the exact numbers. For this problem, you do not need to worry about whether your solution will be fractional or integral.[1]

(a) Give a list of all the variables you will use in your LP, and a few words to describe each (for example, you might say "b_3 is the number of photos given to company b for which they will charge us 3 cents.")

(b) Give a list of constraints.

Update: *We forgot to ask you about the objective function! Feel free to include that information in either part (a) or (b) if it helps you define the LP. Because we didn't correct this bug early on, we won't deduct for forgetting the objective here. You may wish to include/update an objective in the next part as well.*

3.1. There's a Sale!

Company A changed their pricing scheme! They now will charge 5 cents per photo only for your first 100 photos. After that they'll give you a discount. Only 4 cents per photo for your next 300 photos (they still have the same hard cap at 400). How do you solve the problem now? (hint: Robbie's answer involves solving two LPs)

4. Programming: The Dynamically Programmed Textile Factory

Note: Like previous homeworks, this question is due with Homework 6. We still strongly suggest aiming to complete it this week, but want to make sure you have at least a full week of work time. Download the starter code on the course website to get started.

You are given a rectangular piece of cloth with dimensions $X \times Y$, where X and Y are positive integers. You also have a list of n dresses that can be made using the cloth. For the ith dress, you know that a rectangle of cloth of dimensions $a_i \times b_i$ is needed to make it, and that the selling price of the dress is p_i, all of which are positive integers.

[1] Since you're dealing with a very small number of photos, you'd only miss out on a few cents converting a fractional solution to an integer one.

You have a machine that can make a cut any rectangular piece of cloth into two pieces, either horizontally or vertically. Design an algorithm that determines the best return on the $X \times Y$ piece of cloth, that is, a strategy for repeatedly cutting the cloths so that the dresses made from the resulting pieces give the maximum sum of selling prices.

You are free to make as many copies of a given dress as you wish, or none, if so desired. You may rotate the cloth during the manufacturing process (so a 5×3 cloth can make a 3×5 dress). Your solution should use an iterative approach, and run in polynomial time of X, Y and n.