

Assignment 2

Q1 [20 Points] Investment Strategy

Solve problem CLRS 15-10

Q2 [20 Points] Party Planning

You have been assigned as an organizer of a party at the Giggle company and are deciding whom to invite. The company structure is given to you as a rooted tree, with the company president as root and a directed edge (u, v) if u is the supervisor of person v .

Each employee at Giggle is given a (possibly negative) integer rating r measuring how “fun” the employee is. Furthermore, to maximize social interactions between different people, you have been requested to **not** invite an employee and his or her immediate supervisor. Finally, the President of the company (i.e. the root) must attend the party, although the President has a negative fun rating. You would like to invite guests so that the total of the “fun” ratings of the guests is maximized. Using the following steps, design a dynamic programming approach to this problem.

- (a) [3 Points] Clearly define a quantity that a dynamic programming approach computes in English.
- (b) [8 Points] Write a Bellman equation for the quantity defined in part (a) and define the initial conditions. Justify why they work.
- (c) [3 Points] In what order should the Bellman equation in part (b) be computed in a dynamic programming algorithm?
- (d) [3 Points] Analyze the time and space complexity of the algorithm given in parts (b) and (c).
- (e) [3 Points] How would you produce the guest list from the maximum rating computed using the algorithm?

Q3 [20 Points] Doctor Scheduling

A hospital is scheduling doctors for work. The hospital has a desired number of doctors a_1, \dots, a_n that they would like to work on each of the next n days. Furthermore, each doctor who works at the hospital has a list $A_j \subseteq \{1, \dots, n\}$ of days when they would like to work.

(a) [5 Points] You are given the numbers a_1, \dots, a_n and the lists A_1, \dots, A_k of the doctors’ availabilities. Using this data, design an algorithm using network flows to output a schedule S_1, \dots, S_k for each doctor so that:

- Each doctor works only when they are available (i.e. $S_i \subseteq A_i$), and.

- Exactly a_i doctors work on day i .

or the algorithm reports that no schedule is possible given the doctors' availabilities.

(b) [5 Points] Briefly justify the correctness and runtime of the algorithm designed in the previous part (a).

(c) [5 Points] The hospital finds that it may be necessary to schedule doctors for times outside when they report they are available to work. They decide that doctors can be scheduled for up to c days outside their availability list, for some integer $c > 0$. Modify the algorithm designed in part (a) so a schedule S_1, \dots, S_k is produced with:

- S_i containing at most c days not in A_i .
- Exactly a_i doctors work on day i .

or the algorithm reports that no such algorithm is possible.

(d) [5 Points] Briefly justify correctness and runtime of the modified algorithm designed in part (c).

Q4 [10 Points] Exam Invigilation

The University of Toronto assigns Chief Presiding Officers (CPOs) to invigilate final exams for the University. The office in charge of organizing final exams has the following information about exams and CPOs:

- Each exam has a date and a time, where time is one of “morning”, “afternoon” or “evening”. Each combination of dates and times (eg. “the morning of April 14”) is called an examination period.
- Each CPO has a list of examination periods when the CPO is available. CPOs cannot be assigned to slots when they are not available.
- Each CPO has a maximum number of exam periods that the CPO can invigilate, which is less than or equal to the number of periods when the CPO is free.
- Each CPO can invigilate at most two exams on a single day.
- If an examination period has l exams, there should be $\lceil 1.1l \rceil$ CPOs available for that period, in case there are CPOs scheduled who cannot make their shift.

You have been hired by the office to help them assign CPOs to exam periods.

(a) [5 Points] You are provided with a list of the n CPOs and their availabilities, the CPO's work preferences, the m examination slots, and the number of exams in each slot. Design an algorithm using network flow that either assigns each CPO to examination slots satisfying the constraints listed previously, or reports that an assignment is not possible.

(b) [5 Points] Briefly justify the correctness and runtime of the algorithm designed in the previous part, using properties of network flow.