1A Lent Algorithms

Supervision 6: Flows

Hayk Saribekyan

May 3, 2018

When you are asked to describe an algorithm, also analyse its time and space complexity. Mention how you would implement the solution if there are subtle details e.g. order of computation in a tabulation method of DP.

This example sheet is a hybrid of lecturers' problems and mine. Some of the problems added by me are not directly useful for the exams or are out of the scope of the course. These are marked with *. These problems, however, show how some of the standard algorithms that you have learnt can be used. If you have time, you should work on them as they will develop your intuition.

I do not expect you to solve and write down precise solutions to everything, but make sure to read all the problems. You can go back to them during revision or when you have time later. Focus on the concepts that you have most trouble with rather than writing down solutions to problems that are easy for you.

1 Refresher

1. In a city there are n houses, each of which is in need of a water supply. It costs w_i pounds to build a well at house i, and it costs c_{ij} to build a pipe in between houses i and j. A house can receive water if either there is a well built there or there is some path of pipes to a house with a well. Design an algorithm to find the minimum amount of money needed to supply every house with water.

   Hint: Create a vertex that takes the role of all the wells together.

2. * There are n types of boxes. The i^{th} has dimensions $a_i \times b_i \times c_i$ and there are unlimited number of each type. You can stack box A on B if A's base dimensions are *strictly* smaller than B's (i.e. the tower has to get smaller and smaller closer to the top). You are allowed to rotate the boxes. What is the highest box tower you can build?

2 Flows

1. Solve the flow problems (Q8-Q14) from the Lecturer's example sheet 6[1]

2. * Find an example of a graph for which the Ford-Fulkerson algorithm never terminates.

[1] `http://www.cl.cam.ac.uk/teaching/1718/Algorithms/ex6.pdf`

The important thing to notice in this example is that one needs non-rational weights on the edges (try to prove that termination will occur otherwise).

3. In Question 13, how would you find the minimal set of disruptions that will cut off the two stations?

4. You would like to assign n people into m activities in such a way that the largest activity group is as small as possible. For each person, you know exactly which activities they can be part of. Describe an algorithm to find the smallest possible size of the largest group.

   Hint: First, find an algorithm that checks whether, for a given integer s, it is possible to make the assignment in such a way that the largest group has at most s people.

5. * Consider an $n \times m$ matrix, where each element is either 0, 1 or ?. Describe an algorithm that replaces ? with either 0 or 1 such that the sum of elements of row i is a_i, and the sum of elements in column j is b_j. Your algorithm should determine if it is not possible to do so.

   Hint: You need to build some graph and convert the problem into a flow.