Illinois Institute of Technology
Department of Computer Science

Homework Assignment 7

CS 430 Introduction to Algorithms
Spring Semester, 2017

Due: Monday, April 10

1. Suppose you are given an 8-pint jug full of water and two empty jugs of 3- and 5-pint capacity; you must get exactly 4 pints of water in one of the jugs by completely filling/emptying jugs into each other. This problem can be modeled by an implicit graph: The vertices are triples (i, j, k) where i is the amount of water the 3-pint jug, j is the amount of water in the 5-pint jug, and k is the amount of water in the 8-pint jug. The (directed) edges connect a triple T to triples that can be obtained from T by pouring. Thus, for example, there is an edge from $(3, 3, 2)$ to $(1, 5, 2)$ because we can pour from the 3-pint to fill the 5-pint jug, leaving 1 pint in the 3-pint jug.

   (a) You are given jugs of capacities c_1, c_2, and c_3 and have to get a target amount t. Design an algorithm based on breadth-first search to determine how to get t, or to verify its impossibility. Remember, the graph is implicit—you do not have an explicit form of it (in a an adjacency structure, say).

   (b) Justify the correctness of your algorithm (this is just shy of a formal CLRS3-style proof).

   (c) Analyze the time required by your algorithm.
   (*Hint*: See `http://en.wikipedia.org/wiki/Pseudo-polynomial_time`.)

   (d) Implement your algorithm in part (a). The user must be able to enter any three positive integers c_1, c_2, c_3, and a target t. The graph might too big to be kept explicitly. It may be that that target value is impossible to reach and the program must so report. An ideal solution will allow for more than three jugs—say, c_1, c_2, c_3, c_4, and c_5. Your algorithm must give the sequence of pourings to reach the target, when possible, and must indicate when the target cannot be achieved. Your implementation can be in any language and for any platform, but you must turn in hard copy of all the files *and* be prepared to demonstrate your code to the TAs—students will be chosen at random to give such demos.

   For interesting variations of this problem, see chapter 5 of T. S. Michael's *How to Guard an Art Gallery*, Johns Hopkins University Press, 2009.

2. You are helping a group of ethnographers analyze some oral history data they have collected by interviewing members of a village to learn about the lives of people lived there over the last two hundred years. From the interviews, you have learned about a set of people, all now deceased, whom we will denote $P_1, P_2, \ldots, P_n$. The ethnographers have collected several facts about the lifespans of these people, of one of the following

   (a) P_i died before P_j was born.

   (b) P_i and P_j were both alive at some time.

   However, the ethnographers are not sure that their facts are correct; memories are not so good, and all this information was passed down by word of mouth. So, they'd like you to determine whether the data they have collected is at least internally consistent, in the sense that there could have existed a set of people for which all the facts they have learned simultaneously hold. Describe and analyze and algorithm to answer the ethnographer's problem. Your algorithm should either output possible dates of birth and death that are consistent with all the stated facts, or it should report correctly that no such dates exist.

(*Hint*: Construct a bipartite directed graph, in which there $2n$ nodes representing, respectively, dates of birth and death of the n people. Draw edges from one node to another when we have a fact that states that the former precedes the latter. Clearly a person's birth date precedes his/her date of death, so that would be an edge. What does it mean if two people are claimed to have been alive at the same time?)

3. Problem 23-4 on page 641 of CLRS3.