15-451 Homework 5

March 31, 2008

April 8, 2008

Please hand in each problem on a separate sheet and put your **name** and **recitation** (time or letter) at the top of each page. You will be handing each problem into a separate box, and we will then give homeworks back in recitation.

Remember: written homeworks are to be done individually. You must try to solve the problems on your own. If you need to get help from others or from Google, this is acceptable provided that you acknowledge where the ideas came from and write the solution in your own words. You will not lose points for giving proper credit to where your ideas came from.

Direct questions to Bryant Lee (bryantl@cs.cmu.edu)

1 Question 1

[Graph Searching] Let G be a directed graph represented using an adjacency list. So, each node $G[i]$ has a list of all nodes reachable in 1 step from i (all out-neighbors of i). Suppose each node of G also has a value: e.g., node 1 might have value 100, node 2 might have value 50, etc. Give a fast algorithm that computes, for every node, the highest value that can be reached from that node (i.e., that you can get to by some path from that node). For instance, if G is strongly-connected, then for every node this will be the maximum value in the entire graph. Your algorithm should run in time $O(m+n)$ or $O(m+n\log n)$.

2 Question 2

[Graduation] Cranberry-Melon University has n courses. In order to graduate, a student must satisfy several requirements. Each requirement is of the form "you must take at least k courses from subset S". The problem is to determine whether or not a given student can graduate. The tricky part is that any given course cannot be used towards satisfying multiple requirements. For example if one requirement states that you must take at least two courses from $\{A, B, C\}$, and a second requirement states that you must take at least two courses from $\{C, D, E\}$, then a student who had taken just $\{B, C, D\}$ would not yet be able to graduate. Your job is to give a polynomial-time algorithm for the following problem. Given a list of requirements $r_1, r_2, ..., r_m$ (where each requirement r_i is of the form: "you must take at least k_i courses from set S_i"), and given a list L of courses taken by some student, determine if that student can graduate. In particular, show how you can solve this using network flow.

3 Question 3

[Realizing degree sequences] You are the chief engineer for Graphs-R-Us, a company that makes graphs to meet all sorts of specifications.

1. A client comes in and says he needs a 4-node directed graph in which the nodes have the following in-degrees and out-degrees:

   $d_{1,in} = 0, d_{1,out} = 2$
   $d_{2,in} = 1, d_{2,out} = 2$
   $d_{3,in} = 1, d_{3,out} = 1$
   $d_{4,in} = 3, d_{4,out} = 0$

   Is there a directed graph, with no multi-edges or self loops, that meets this specification? If so, what is it? If not, why not?

2. This type of specification, in which the in-degrees and out-degrees of each node are given, is called a degree sequence. The question above is asking whether a given degree sequence is realizable that is, whether there exists a directed graph having those degrees.

   Find an efficient algorithm that, given a degree sequence, will determine whether this sequence is realizable, and if so will produce a directed graph with those degrees. The graph should not have any self-loops, and should not have any multi-edges (i.e., for each directed pair (i, j) there can be at most one edge from i to j, though it is fine if there is also an edge from j to i). Hint: as if you couldn't have guessed - think network flow!