Computer Science C73 November 18, 2019
Scarborough Campus University of Toronto

Homework Assignment #8
(worth 4% of the course grade)
Due: December 2, 2019, by 10 am

• ***You must submit your assignment through the Crowdmark system.*** *You will receive by email an invitation through which you can submit your work in the form of separate PDF docucments with your answers to each question of the assignment. To work with a partner, you and your partner must form a group on Crowdmark. Crowdmark does not enforce a limit on the size of groups.* ***The course policy that limits the size of each group to at most two remains in effect:*** *submissions by groups of more than two persons will not be graded.*
• *It is your responsibility to ensure that the PDF files you submit are legible. To this end, I encourage you to learn and use the LaTex typesetting system, which is designed to produce high-quality documents that contain mathematical notation. You are not required to produce the PDF files you submit using LaTex; you may produce it any way you wish, as long as the resulting document is legible.*
• *By virtue of submitting this assignment you (and your partner, if you have one) acknowledge that you are aware of the policy on homework collaboration for this course.*[a]
• *For any question, you may use data structures and algorithms previously described in class, or in prerequisites of this course, without describing them. You may also use any result that we covered in class, or is in the assigned sections of the official course textbooks, by referring to it.*
• *Unless we explicitly state otherwise, you may describe algorithms in high-level pseudocode or point-form English, whichever leads to a clearer and simpler description.*
• *Unless we explicitly state otherwise, you should justify your answers. Your paper will be graded based on the correctness and efficiency of your answers, and the clarity, precision, and conciseness of your presentation.*

[a] "In each homework assignment you may collaborate with at most one other student who is currently taking CSCC73. If you collaborate with another student on an assignment, you and your partner must submit only one copy of your solution, with both of your names. The solution will be graded in the usual way and both partners will receive the same mark. Collaboration involving more than two students is not allowed. **For help with your homework you may consult only the instructor, TAs, your homework partner (if you have one), your textbook, and your class notes. You may not consult any other source.**"

Question 1. (10 marks) Given below are three variations on the maximum-flow problem. Show that each can be solved in polynomial time, by reducing it to a linear program. Explain the role of the variables, the objective function, and the constraints of your linear program.

a. Maximum flow with leaky nodes. We are given a flow network (G, s, t, c) with graph $G = (V, E)$, source and sink nodes s and t, respectively, and capacities $c(e)$ for each edge $e \in E$. In addition, for each node $u \in V - \{s, t\}$, we are given a constant ℓ_u, $0 \leq \ell_u \leq 1$, that represents the "leakage" that occurs at the node: that is, the outgoing flow from u is only a fraction ℓ_u of the incoming flow into u. We want to find the maximum amount of traffic that can arrive at t, assuming that an unlimited amount of traffic can be generated at s.

b. Minimum cost flow. We are given a flow network (G, s, t, c) with graph $G = (V, E)$, source and sink nodes s and t, respectively, and capacities $c(e)$ for each edge $e \in E$. In addition we are given a desired amount of flow d, and the price $p(e)$ of shipping one unit of traffic on edge e. We want to find the minimum cost of shipping a total flow of d units of traffic from s to t.

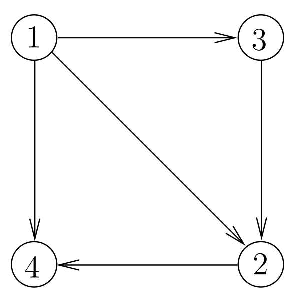
c. Multi-product flow. We are given a graph $G = (V, E)$, and k "products". Each product i, $1 \le i \le k$, has a source $s_i \in V$ (the node where this product is generated), and a destination $t_i \in V$ (the node where this product is consumed), where $s_i \neq t_i$. Product i must be shipped from s_i to t_i along one or more paths of G, subject to the capacity and conservation constraints: An edge cannot carry more traffic than its capacity, what goes into a node must come out of it or be consumed by it (if the node is the destination of some product(s)); and what comes out of of a node must have come into it or have been generated by it (if the node is the source of some product(s)). Note that more than one product may be shipped through any edge. We wish to maximize the total amount of traffic of all products that are shipped from their sources to their destinations.

Question 2. (15 marks) Let $G = (V, E)$ be a directed graph. For each node u, the ***in-degree*** (respectively, ***out-degree***) of u in G is the number of edges in E that go into (respectively, out of) u. For the remainder of this question we consider only directed graphs with no self-loops, i.e., no edge from a node to itself.

Suppose that we are given a set of nodes V (but no edges), and for each $u \in V$, a pair of non-negative integers, $in(u)$ and $out(u)$; we call this the ***degree-pair*** of u. Is there a directed graph $G = (V, E)$ such that for each $u \in V$, the in-degree of u in G is $in(u)$ and the out-degree of u in G is $out(u)$? If so, we say that the given degree-pairs are ***realizable***, and that the graph G ***realizes*** them.

For example, the degree pairs given below for $V = \{1, 2, 3, 4\}$ is realizable, as confirmed by the graph to the right.

u	$in(u)$	$out(u)$
1	0	3
2	2	1
3	1	1
4	2	0



On the other hand, the degree pairs given below for $V = \{1, 2, 3\}$ is not realizable: since each of nodes 2 and 3 has out-degree 2, each of must have an edge to the third node 1, whose in-degree, however, is only 1.

u	$in(u)$	$out(u)$
1	1	1
2	2	2
3	2	2

a. Give a zero-one linear program that has a solution if and only if a given set of degree pairs is realizable; and if it is, explain how to construct a graph that realizes the given degree pairs from the solution to your 0-1 LP.

b. Give a polynomial-time algorithm that, given a set of nodes V and a pair of non-negative integers $\big(in(u), out(u)\big)$ for each $u \in V$, determines whether these degree-pairs are realizable. If they are, the algorithm should also output a directed graph that realizes them. Explain why your algorithm works, and analyze its running time.

Hint. Reduce this problem to a max-flow problem: Define a flow network $\mathcal{F} = (G, s, t, c)$ that, in addition to source and target nodes, has two nodes, v_{out} and v_{in}, for each node $v \in V$. It is up to you to determine the edges of G and their capacities. G should be defined in such a manner that the given degree pairs are realizable if and only if the maximum flow of $\mathcal{F}$ satisfies a certain (easy-to-check) property.

THAT'S IT WITH HOMEWORK, FOLKS!