CSOR W4246 – Fall, 2018

Homework 3 – Theoretical part

Out: Friday, October 26, 2018
Due: 8pm, Friday, November 9, 2018

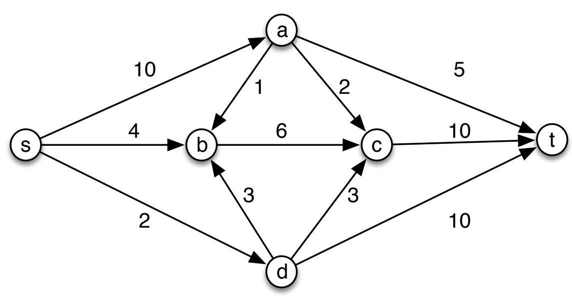
Please keep your answers clear and concise. For all algorithms **you** *suggest, you must prove correctness and give the best upper bound that you can for the running time. You should always describe your algorithm clearly in English* **and** *give pseudocode. If you give a reduction or a linear program, you should follow the steps discussed in class.*

You should not use any external resources *for this homework. You should write up the solutions* **entirely on your own**. *Collaboration is limited to discussion of ideas only. Similarity between your solutions and solutions of your classmates or solutions posted online will result in receiving a 0 in this assignment and possibly further disciplinary actions.*

You must submit your assignment as a pdf file. *Other file formats, such as jpg, doc, c, or py, will not be graded, and will automatically receive a score of 0. If you do not type your solutions, be sure that your hand-writing is legible, your scan is high-quality and your name is clearly written on your homework.*

If you do not type your solutions, make sure that your hand-writing is legible, your scan is high-quality and your name is clearly written on your homework.

1. (25 points) Run the Ford-Fulkerson algorithm on the following network, with edge capacities as shown, to compute the max s-t flow. At every step, draw the residual graph and the augmenting paths. Report the maximum flow along with a minimum cut.



2. (25 points) A *flow network with demands* is a directed capacitated graph with potentially multiple sources and sinks, which may have incoming and outgoing edges respectively. In particular, each node $v \in V$ has an integer *demand* d_v; if $d_v > 0$, v is a *sink*, while if $d_v < 0$, it is a *source*. Let S be the set of source nodes and T the set of sink nodes.

   A *circulation with demands* is a function $f : E \to R^+$ that satisfies

   (a) *capacity constraints:* for each $e \in E$, $0 \leq f(e) \leq c_e$.

   (b) *demand constraints:* For each $v \in V$, $f^{\text{in}}(v) - f^{\text{out}}(v) = d_v$.

   We are now concerned with a decision problem rather than a maximization one: *is there* a circulation f with demands that meets both capacity and demand constraints?

   i. Derive a necessary condition for a feasible circulation with demands to exist.

   ii. Reduce the problem of finding a feasible circulation with demands to Max Flow.

3. (25 points) Similarly to a flow network with demands, we can define a *flow network with supplies* where each node $v \in V$ now has an integer *supply* s_v so that if $s_v > 0$, v is a *source* and if $s_v < 0$, it is a *sink*, and the supply constraint for every $v \in V$ is $f^{\text{out}}(v) - f^{\text{in}}(v) = s_v$.

   In a *min-cost flow* problem, the input is a flow network with supplies where each edge $(i, j) \in E$ also has a cost a_{ij} (per unit of flow). Given a flow network with supplies and costs, the goal is to find a feasible flow $f : E \to R^+$ —that is, a flow satisfying edge capacity constraints and node supplies—that minimizes the total cost of the flow.

   (a) Show that max flow can be formulated as a min-cost flow problem.

   (b) Formulate a linear program for the min-cost flow problem.

4. (25 points) A paper mill manufactures rolls of paper of standard width 3 meters. Customers want to buy paper rolls of shorter width, and the mill has to cut such rolls from the 3m rolls. For example, one 3 m roll can be cut into 2 rolls of width 93cm and one roll of width 108cm; the remaining 6cm goes to waste. The mill receives an order of

   - 97 rolls of width 135cm
   - 610 rolls of width 108cm
   - 395 rolls of width 93cm
   - 211 rolls of width 42cm

   Form a linear program to compute the smallest number of 3m rolls that have to be cut to satisfy this order, and explain how they should be cut.

RECOMMENDED exercises: *do NOT return, they will not be graded.*

1. There are many variations on the maximum flow problem. For the following two natural generalizations, show how to solve the more general problem by **reducing** it to the original max-flow problem (thereby showing that these problems also admit efficient solutions).

   - There are multiple sources and multiple sinks, and we wish to maximize the flow between all sources and sinks.
   - Both the edges *and the vertices* (except for s and t) have capacities. The flow into and out of a vertex cannot exceed the capacity of the vertex.