max planck institut informatik

SAARLAND UNIVERSITY
COMPUTER SCIENCE

Karl Bringmann, Erik Jan van Leeuwen **Winter 2016/2017**

Exercises for Algorithms and Data Structures

`http://www.mpi-inf.mpg.de/departments/algorithms-complexity/teaching/winter16/algorithms-and-data-structures/`

Exercise Sheet 8 Due: **9.1.2017**

The homework must be handed in on Monday before the lecture. You may collaborate with other students on finding the solutions for this problem set, but every student must hand in a writeup in their own words. We also expect you to state your collaborators and sources (books, papers, course notes, web pages, etc.) that you used to arrive at your solutions.

You need to collect at least 50% of all points on all exercise sheets to be admitted to the final exam.

Whenever you are asked to design an algorithm in this exercise sheet you have to give a proof of its correctness as well as an asymptotic upper bound on its worst case running time.

Exercise 1 (*10 points*) Execute the Ford-Fulkerson algorithm on the graph G given in Figure 1, and determine a maximum-value flow (the flow and its value) and a minimum-capacity cut (the cut and its capacity). For every iteration of the algorithm, draw the residual graph and show the flow augmenting path and the amount of flow that is used to augment the previous flow.

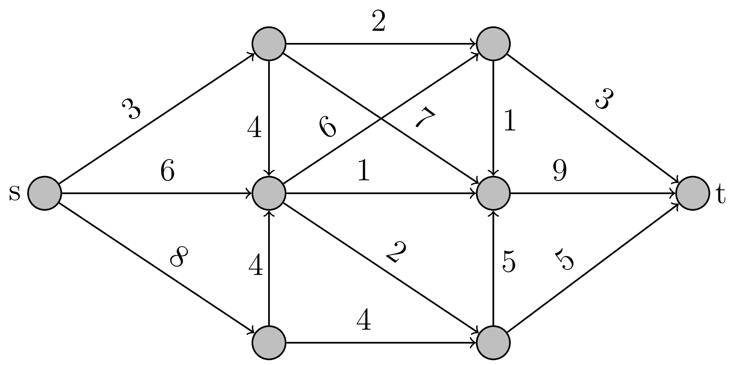

Figure 1: Input graph G for Exercise 1. The number next to each edge denotes its capacity.

Exercise 2 (*10 points*) A car manufacturer runs m factories (they all produce the same car) and n car dealerships. Each factory i is able to manufacture s_i cars per month. Every dealer

j determines a number d_j of cars that he thinks he will sell next month. The manufacturer is able to transport $c_{i,j}$ cars every month from factory i to dealer j. The car manufacturer is concerned if the m factories can handle the demand of the n car dealers and asks you to design an algorithm that checks for the upcoming month whether or not there is a way to distribute enough cars from the factories to the dealerships.

Design and analyze an algorithm for this problem. You should use a maximum flow algorithm as a subroutine.

Exercise 3 (*10 points*) Let $D = (V, A)$ be a directed graph with a capacity function $c : A \to \mathbb{R}$, and let $s, t \in V$.

a) (4 points) Suppose that D has parallel arcs (i.e. there exist vertices $u, v \in V$ such that $(u, v), (v, u) \in A$). Describe and analyze an algorithm that constructs a graph D' without parallel arcs that has the same minimum s-t cut capacity as D. Argue how a minimum s-t cut in D' can be transformed to a minimum s-t cut in D and vice versa. You are *not* allowed to use any maximum flow algorithm.

b) (6 points) Suppose that we are additionally given a capacity function $c' : V \to \mathbb{R}$, and we impose the additional constraint on any flow f (i.e. beyond the usual ones) that $\sum_{a \in \delta^{in}(v)} f(a) \leq c'(v)$ for all $v \in V \setminus \{s, t\}$. Describe and analyze an algorithm to find a maximum flow under these conditions.

   Hint: you can use the maximum flow algorithm as a black box.

Exercise 4 (*10 points*) Let $D = (V, A)$ be a directed graph with a capacity function $c : A \to \mathbb{N}$, and let $s, t \in V$. Given a valid s-t flow f, let D_f denote the residual graph with respect to f.

Now suppose that f is a valid s-t flow, and let P be a shortest s-t path in D_f. Let f' be the flow after augmenting it using P.

a) (3 points) Prove that $D_{f'}$ contains P^{-1} (i.e. the reverse arc of every arc of P), and that there is at least one arc of P that is not in $D_{f'}$.

b) (3 points) Show that any s-t path in $D_{f'}$ is at least as long as P.

c) (2 points) Show that if a shortest s-t path in $D_{f'}$ is equally long as P, then the number of arcs that is contained in at least one shortest s-t path is less in $D_{f'}$ than it is in D_f.

d) (2 points) Use the preceding observations to describe and analyze an $O(nm^2)$ time algorithm for maximum s-t flow.