# Tutorial 3

1. Let G be a network with source s, sink t, and integer capacities. Prove or disprove the following statements:

   (a) If all capacities are even then there is a maximal flow f such that f(e) is even for all edges e.

   (b) If all capacities are odd then there is a maximal ow f such that f(e) is odd for all edges e.

2. In some country there are n cities and m bidirectional roads between them. Each city has an army. Army of the i-th city consists of $a_i$ soldiers. Now soldiers roam. After roaming each soldier has to either stay in his city or go to the one of neighboring cities by moving along at most one road. Check if it is possible that after roaming there will be exactly $b_i$ soldiers in the i-th city.

3. You have seen how to find edge-connectivity of a graph using max-flow. Suppose that we want to find not only the minimum number of edges that disconnect the graph, but also the exact set of edges. Can you modify the algorithm you studied to do that?

4. A number k of trucking companies, c1,...,ck, want to use a common road system, which is modeled as a directed graph, for delivering goods from source locations to a common target location. Each trucking company $c_i$ has its own source location, modeled as a vertex $s_i$ in the graph, and the common target location is another vertex t. (All these k +1 vertices are distinct.) The trucking companies want to share the road system for delivering their goods, but they want to avoid getting in each other's way while driving. Thus, they want to paths in the graph, one connecting each source $s_i$ to the target t, such that no two trucks use a common road. We assume that there is no problem if trucks of different companies pass through a common vertex. Design an algorithm for the companies to use to determine k such paths, if possible, and otherwise return "impossible".

5. Prove that every k-regular bipartite graph has a perfect matching (prove using notions of max flow).

6. How fast can you find the maximum matching in a bipartite graph using maximum flow? (Use push-relabel in specific order)

7. Suppose that instead of a single capacity c(u,v) on each edge, each edge has a pair (l(u,v), c(u,v)) and the capacity constraint is changed such that the l(u,v) <= f(u,v) <= c(u,v) (So each edge must carry some minimum flow on it). Can you find a feasible flow on this network? If yes, can you then find the maximum flow?