1. Suppose we have \( n \) currencies \( c_1, \ldots, c_n \) among which we can make trades. For example, \( c_1 \) is euro, \( c_2 \) pound sterling, \( c_3 \) US dollar etc. For each pair \((c_i, c_j)\) there is an exchange rate \( r_{ij} \) for buying currency \( c_j \) using currency \( c_i \). That is, \( x \) units of \( c_i \) buys you \( r_{ij} \cdot x \) units of \( c_j \). We assume that for all \( i \) and \( j \) the rate \( r_{ij} \) is positive and finite.

Usually there are costs associated with currency trading, so we would expect that for example \( r_{ij} r_{ji} < 1 \) for all \( i, j \). However, occasionally for a short period of time a situation may exist where some currencies \( c_{i_1}, \ldots, c_{i_k} \) satisfy

\[
r_{i_1 i_2} r_{i_2 i_3} \cdots r_{i_{k-1} i_k} r_{i_k i_1} > 1.
\]

This means that funds we have in currency \( c_{i_1} \) can be increased for free by rotating it via currencies \( c_{i_2}, \ldots, c_{i_k} \). Give an algorithm for detecting such situations, when the exchange rates \( r_{ij} \) are given.

2. There is nothing in the definition of minimum spanning tree that would require the edge weights to be positive.

(a) We wish to find the minimum spanning tree for a graph \( G \) that contains some edges with negative weights. Suppose we have access to a procedure \( P \) that solves the minimum spanning tree problem but works only for graphs without negative weights. How can we easily modify \( G \) so that we can use \( P \) to find its minimum spanning tree? \textit{Hint:} all spanning trees have the same number of edges.

(b) How can you find the spanning tree with \textit{maximum} weight using an algorithm for minimum spanning trees?

3. A railroad network is given as a graph \( G = (V, E) \), where the vertices represent stations and edges represent direct rail connections. For each rail connection \((u, v) \in E\), there is a limit \( w(u, v) \) for how heavy shipments the connection can safely handle. Give an algorithm to find the largest weight \( W \) such that from any starting station \( u \), a shipment of weight \( W \) can be routed safely to any destination \( v \).

4. We have a set of variables \( x_1, \ldots, x_n \), and for them a set of \textit{equality constraints} of the form \( x_i = x_j \) and \textit{inequality constraints} of the form \( x_i \neq x_j \). Give an efficient algorithm do decide, whether all the constraints can be satisfied simultaneously.

5. Please help us further develop the course by filling in the feedback form found at


Notice that the feedback is in principle anonymous, but due to the small number of foreign students this may in practice not hold for answers given in English.