First set of obligatory problems for the course 'Graph algorithms with practical applications' (DM201)

Department of Mathematics and Computer Science
University of Southern Denmark

The problems are handed out Tuesday March 11. 2008. The solutions must be returned by Tuesday April 15. 2008

It is important that you explain how you obtain your answers and argue why they are correct. If you are asked to describe an algorithm, then you must supply enough details so that a reader who does not already know the algorithm can understand it. However, in general it is not necessary to write a long pseudocode. You may explain what the algorithm does in words. Note also that illustrating an algorithm means that one has to follow the steps of the algorithm meticulously (slavisk). Notation and references to results refer to the course book "Digraphs: Theory, Algorithms and Applications" by Bang-Jensen and Gutin.

The reports are evaluated pass/fail by the teacher. In order to pass you should demonstrate that you can work satisfactory with the topics dealt with. In particular, you must attempt a solution of all problems. On the other hand, you will pass on less than a perfect solution. You may work in groups of up to 3 persons. Notice that we may ask questions regarding your solution at the oral exam.

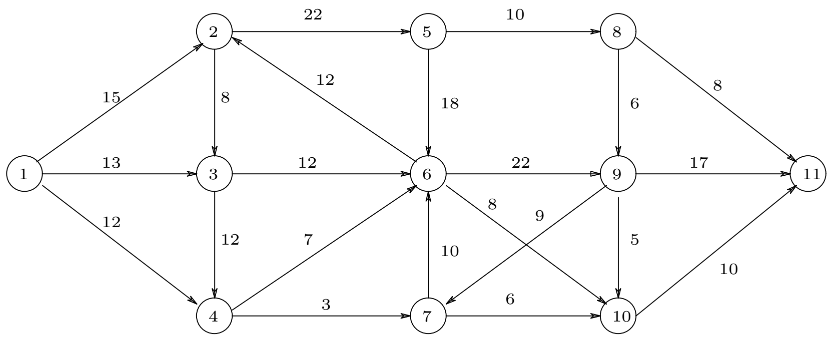

Figure 1: A network with capacities shown and all lower bounds zero. The source is the vertex 1 and the sink is the vertex 11.

PROBLEM 1

Question a:

Give a short description of the capacity scaling algorithm for finding a maximum (s,t)-flow in a network and illustrate the algorithm by applying it to the network in Figure 1

Question b:

Give a short description of Dinic's algorithm for finding a maximum (s,t)-flow in a network and illustrate the algorithm by applying it to the network in Figure 1

Question c:

Suppose now that we change the network above by giving each arc a lower bound of 1. Show how to find a minimum value feasible flow in the resulting network. Remember to explain the general method that you apply (that is, it is not sufficient to just show a feasible flow in the network).

PROBLEM 2

In this problem we study arc-connectivity of directed graphs. A spanning subdigraph of a digraph $D = (V, A)$ is a digraph $D' = (V, A')$ such that $A' \subseteq A$. That is, D' has the same vertices as D and is obtained from D by deleting zero or more arcs. (In particular, the digraph $D_\emptyset = (V, \emptyset)$ where we have deleted all arcs is a spanning subdigraph of D. Note also that if D' is a strong spanning subdigraph of $D = (V, A)$ and $|V| \geq 2$, then every vertex of V has at least one arc entering and at least one arc leaving in D'.)

Let $\lambda(D)$ denote the maximum integer k such that D is k-arc-strong. Below we always assume that $k = \lambda(D)$. An arc $a = s \to t$ is **critical** (for k-arc-strong connectivity) if $\lambda(D - a) < k$. Here $D - a$ is the digraph obtained from D by deleting the arc a.

A digraph $D = (V, A)$ is minimally k-arc-strong if $\lambda(D) = k$ but $\lambda(D - a) < k$ for every $a \in A$. That is, every arc in A is critical.

Question a:

Show that $a = s \to t$ is critical if and only if there exists a set X with $s \in X$ and $t \in V - X$ and $d^+_D(X) = k$.

Question b:

Show how to decide in time $O(k|A|)$ whether a given arc a is critical in a digraph $D = (V, A)$. Hint: use flows.

Question c:

Describe an $O(k|A|^2)$ algorithm to find a minimally k-arc strong spanning subdigraph of a given input digraph $D = (V, A)$. Remember to argue that your algorithm is correct.

Question d:

Does every digraph have a unique minimally k-arc strong spanning subdigraph?

Question e:

Suppose $\lambda(D) = k$ and that $a = s \to t$ is a critical arc of D. Prove that in $D - a$ there exists a unique minimal set (with respect to inclusion) X such that $s \in X$, $t \in V - X$ and $d^+_{D-a}(X) = k - 1$ and a unique minimal set Y such that $t \in Y$, $s \in V - Y$ and $d^-_{D-a}(X) = k - 1$.

PROBLEM 3

This problem concerns matroid intersection and the algorithm described in Korte and Vygen section 13.5. A **multigraph** is a graph in which we allow parallel edges.

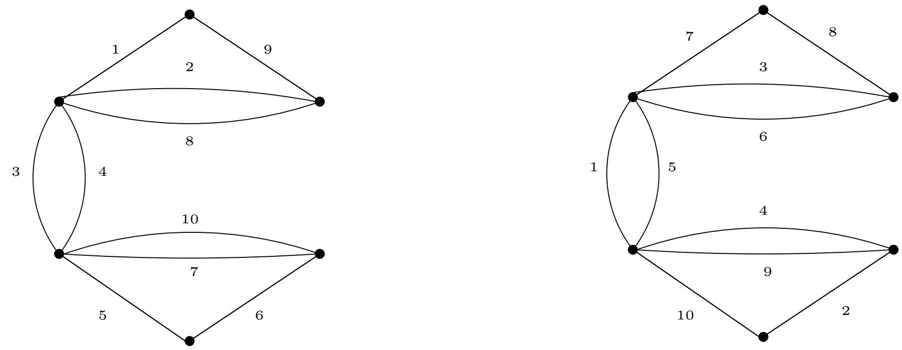

Figure 2: Two multigraphs and the relation between their edges (the numbers illustrate the mappings f_1, f_2). G_1 is the left graph and G_2 the right one.

Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two multigraphs with the same number m of edges and let $f_i : E_i \to \{1, 2, \ldots, m\}$, $i = 1, 2$, be bijections. Extend f to subsets of E_i in the obvious way: $f_i(E'_i) = \{f_i(e) : e \in E'_i\}$.

Consider the problem of finding a maximum cardinality set E' of edges in G_1 so that E' induces a forrest (contains no cycles) in G_1 and $f_2^{-1}(f_1(E'))$ induces a forrest in G_2. Equivalently: we are looking for a maximum cardinality subset S of $\{1, 2, \ldots, m\}$ so that the edges with numbers from S form a forrest in G_i for $i = 1, 2$.

Question a:

Show how to formulate this problem as a matroid intersection problem. Hint: you may use $\{1, 2, \ldots, m\}$ as the ground set for both matroids.

Question b:

Let G_1, G_2 be the graphs in Figure 2 where the number on the edges in G_i gives the functions f_i, $i = 1, 2$, i.e. the numbering of the edges. Argue that with these numberings the images (under f_i^{-1}, $i = 1, 2$) of the set $X = \{1, 2, 3, 10\}$ is a forrest in both graphs but no further element can be added to X while preserving this property in both graphs.

Question c:

Draw the graph G_X (described on page 297 in Korte) and describe how you obtain its arc set.

Question d:

Describe a directed path in G_X which shows that X is not a maximal common independent set of your matroids and give the resulting common spanning tree in both graphs.

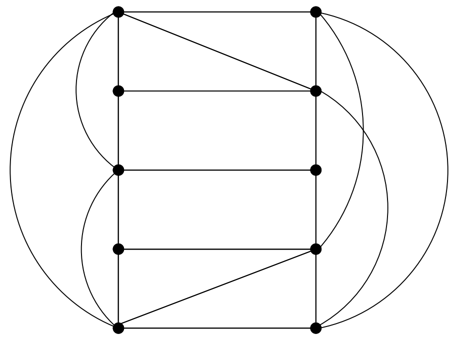

Figure 3: A graph G.

PROBLEM 4

This problem is based on the method described in the handout notes from week 9 on maximum adjacency orderings. They are available from the course page.

Let $G = (V, E)$ be the graph in Figure 3.

Question a:

Show how to determine the edge-connectivity $\lambda(G)$ by using consequtive maximum adjacency orderings and contractions.

Question b:

Show a sparse certificate for the $\lambda(G)$-edge-connectivity of G which you obtain from one maximum adjacency ordering.