Klausur Optimierung B

Sommersemester 2013, Prof. Dr. Ir. Arie M.C.A. Koster,
Assistent: Dipl.-Comp.Math. Stephan Lemkens (Lehrstuhl II für Mathematik)

Anmerkung zur Klausur:

Der Autor dieses Dokuments hat selbst nicht an der folgenden Klausur teilgenommen. Stattdessen wurde diese Klausur als Probeklausur mit Lösungen im Wintersemester 13/14 zur Verfügung gestellt. Die Aufgaben wurden in der eigentlichen Prüfung auf Englisch gestellt, wobei eine deutsche Übersetzung beilag. Man sollte anmerken, dass die Klausur nach Informationen des Assistenten schlecht ausgefallen ist.
Der Autor übernimmt keinerlei Verantwortung bezüglich Korrektheit oder Vollständigkeit der Aufgaben bzw. Lösungen.

Task 1) (15 points)

Which of the following statements is true? Give a proof for the statement or a counterexample. Let $N = (V, A)$ be a network with a source $s \in V$ and a sink $t \in V$. Furthermore, let $c : A \longrightarrow \mathbb{Z}_+$ be a corresponding capacity function.

a) If $f : A \longrightarrow \mathbb{R}$ is a maximum s-t-flow for N, either $f(u,v) = 0$ or $f(u,v) = c(u,v)$ holds for every arc $(u,v) \in A$

b) A maximally augmenting path for an s-t-flow has a minimal number of arcs compared to all other augmenting paths.

c) If every capacity is multiplied by a number $\lambda \in \mathbb{R}_+$ every minimal s-t-cut of the original graph remains a minimal s-t-cut in the modified network.

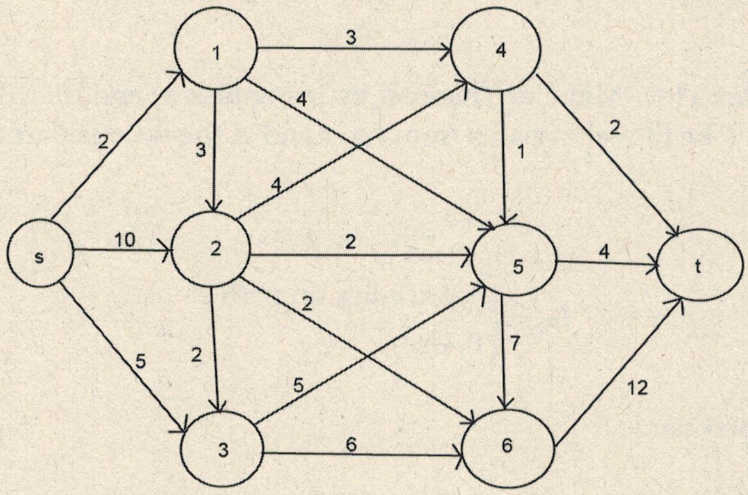
Let G be a graph. Consider the following definitions:

$\tau(G) := \min\{|W| : \text{W is a vertex cover in G}\}$
$\nu(G) := \max\{|M| : \text{M is a matching in G}\}$
$\rho(G) := \min\{|F| : \text{F is an edge cover in G}\}$

d) For each acyclic graph G without isolated vertices it holds: $|V| - \rho(G) < \nu(G)$

e) For each graph without isolated vertices it holds: $\nu(G) = \tau(G)$

Task 2) (6 points)
Compute a maximum s-t-flow and a minimum s-t-cut in the following network D. The values on the arcs denote the corresponding capacities.



Task 3 (4 Points)
Let $a \in V$ be a node in a connected and simple graph $G = (V, E)$. Show: The graph G is bipartite if and only if the length of a shortest a-v-path is not equal to the length of a shortest a-w-path for each edge $vw \in E$.

Task 4 (3 Points)
A bipatite graph $G = (V, E)$ has a perfect matching if and only if a vertex cover contains at least $\frac{1}{2}\ |V|$ nodes.

Task 5 (4 points)
An undirected graph $G = (V, E)$ is called 2-connected if $G - e$ is connected for each edge $e \in E$. Proof the following: A connected graph $G = (V, E)$ is 2-connected if and only if each $e \in E$ is part of a cycle.

Task 6 (3+4+3 Points)
Consider the connection between the max flow and the min cut in the context of linear programming. Let $D = (V, A)$ be a network with source s and sink t and let c be the corresponding capacity function.

a) Formulate the max-flow-problem as a linear programm. In your model each variable should describe the value of the flow on a path from s to t.

b) Determine the dual of the linear programm modeled in task a) and interpret the resulting problem.

c) Proof the Max-Flow-Min-Cut Theorem by using task a) and b).
Hint for c: Let P be the set of paths from s to t and A the set of all arcs. The $|P| \times |A|$-matrix I with

$$I_{Pa} = \begin{cases} 1, \text{arc } a \text{ lies on path } P \\ 0, \text{else} \end{cases}$$

is totally unimodular.

Task 7 (2+2+4 Points)
Given an undirected graph $G = (V, E)$ with positiv node weights. The Max Weighted Independent Set Problem consists of finding an independent set in G with maximal total weight. In the following consider the case where G is a path.

a) Proof, that the following algorithm does not always compute an optimal solution:

Choose $S = \emptyset$ and $C = V(G)$
while C contains nodes choose the node v_i with maximal weight and add v_i to S
remove v_i and all its neighbours from C
return S

b) Assume the nodes $v_1, ..., v_n$ of G are sorted in such a way that the edges only exist between nodes v_i and v_{i+1} with $1 \leq n-1$. Proof that the following algorithm does not always compute an optimal solution:

Let S_1 be the set of all $v_i \in G$ with i odd
Let S_2 be the set of all $v_i \in G$ with i even
Return the S_i with greater total weigth.

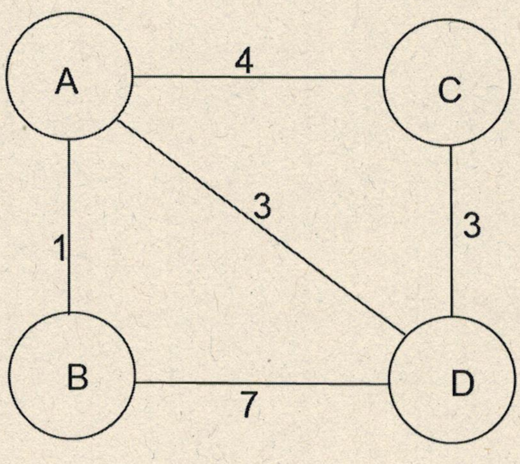
c) State a dynamic programm which always computes the optimal solution. Given a weighted path with n nodes, your algorithm should have a running time polynomial in n.

Task 8) (2+2+6 Points)

Let $G = (V, E)$ be an undirected graph with edge weights $w_e \geq 0$. The following partitioning of the graph is called a k-node-partition of G:

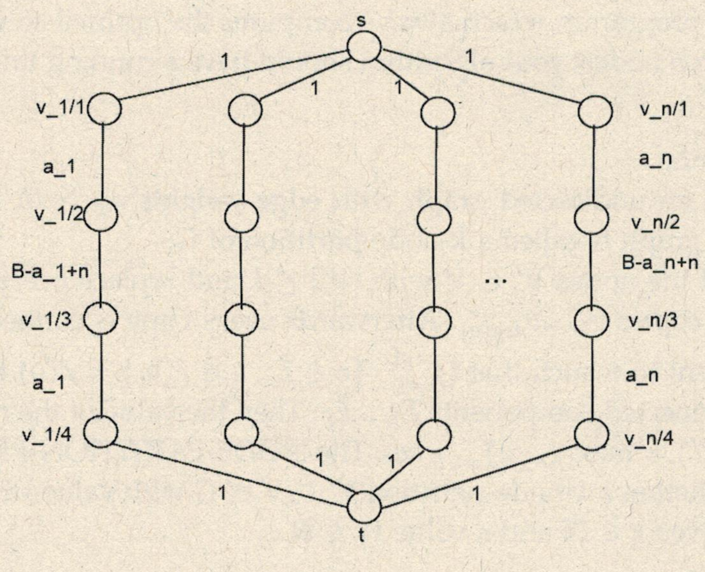
Choose a subset of the nodes $V' \subset V$ with $|V'| \leq k$ and replace every node $v \in V'$ with deg(v) many copies $v_1, ...v_{deg(v)}$. Afterwards every copy is connected with one of the nodes incident to v such that $\bigcup_{i=1}^{deg(v)} \{e \in E : e \in \delta(v_i\} = \delta(v)$ holds. We call the sets of the S connected components $E_1, ...E_S$. Then the value of the node-partition is given by $W_{max}(V') = \max_{1 \leq s \leq S} \sum_{e \in E_s} w_e$. The NODE-PARTITION-PROBLEM consists of deciding whether a k-node-partition $V' \subset V$ of G with value smaller or equal to W exists for a given $k \in \mathbb{N}$ and a value $W \in \mathbb{R}_+$.

a) Determine a minimal 2-node-partition of the following graph.



b)What are the properties a graph must have such that a non-trivial 1-node-partition exists?

c)The proof that NODE-PARTITION is NP-complete in general is done by a reduction from Partition. Consider positive integer values $a_1, ..., a_n$ and an integer $B = \sum_{i=1}^{n} \frac{a_i}{2}$. The weighted graph G necessary for the reduction is given by:



Proof the following main idea of the reduction: The set $A = \{a_1, ..., a_n\}$ can be partitioned into two sets with sum B if and only if G contains a 2n-node-partition with value $B + n$.