Combinatorische Optimalisering, Exam 29 March 2012

Het tentamen duurt 3 uur. Studenten die inzage in hun resultaten willen nadat de cijfers bekend zijn kunnen daartoe een afspraak maken per e-mail.

Het tentamen bestaat uit 4 opdrachten. Het maximum aantal punten dat verdient kan worden voor de verschillende onderdelen staat in de volgende tabel:

1	2a	2b	2c	2d	2e	3a	3b	3c	4a	4b
10	10	10	10	10	5	10	5	10	10	10

Het totale maximum aantal punten 100. Het cijfer wordt verkregen door dit aantal door 10 te delen, met een ondergrens van 1. Dus 55 punten zijn nodig om te slagen.

Gedurende het tentamen mag alleen het boek Comb. Opt. by Pap. & Steigl. *zonder losse blaadjes erin op tafel liggen en alle electronische apparaten moeten uitgeschakeld worden.*

---

1. Decide for each of the following statements if it is "true" or "false" without giving any explanation. A correct answer gives 1 point, a mistake gives −1 point. No answer gives 0 points. In the statements a graph refers always to a simple graph, thus not to a multi-graph in which parallel edges are allowed.

(a) Every complete bipartite graph is regular.
Hint: Remember that a regular graph is a graph with vertices that have all the same degree.

(b) A graph is regular if and only if its complement is regular.

(c) A 5-regular graph on 16 vertices has 80 edges.

(d) If in a graph there exists a walk from vertex s to vertex t then there exists a path from s to t.

(e) A graph of which each vertex has degree at least $k-1$, has a path of length k.

(f) There exists a non-connected graph with 6 vertices and 11 edges.

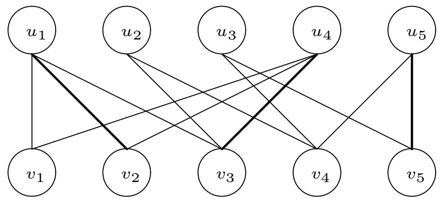
(g) A spanning forest of a graph on n vertices consisting of k components has $n-k-1$ edges.

(h) There exist 20 different Hamilton circuits in K_5 (the complete graph on 5 vertices).

(i) There exists a graph with 5 vertices, 3 of which have degree 3.

(j) If a graph has n vertices and m edges and $n > m$ then the graph has at least $n-m$ components.

2a. Given is a bipartite graph $G = (U, V)$. The U-vertices are numbered $u_1, u_2, \ldots, u_5$ and the V-vertices are numbered $v_1, v_2, \ldots, v_5$. The boldface printed edges $\{u_1, v_2\}$, $\{u_4, v_3\}$,$\{u_5, v_5\}$ form a matching of cardinality 3 (sorry for not having been able to make them clearer boldface in the picture). Apply the method of augmenting paths to find a perfect matching in G. Show clearly how you find augmenting paths.



2b. Given a complete bipartite graph on two times 3 vertices. The weights of the edges are given in the following table.

	v_1	v_2	v_3
u_1	9	8	2
u_2	7	6	0
u_3	6	3	1

Use the Hungarian Method to find the minimum weight perfect matching in this graph. Show the steps that lead you to the optimal solution.

2c. Given a graph $G = (V, E)$ and a subset of the vertices $V' \subset V$, we define $\Gamma(V')$ as the set of all vertices not in V' that are adjacent to some vertex in V':

$$\Gamma(V') = \{v \in V \setminus V' \mid \exists u \in V' \text{ with } \{u, v\} \in E\}.$$

Let $|X|$ denote the cardinality of X, i.e. the number of elements in the set X.

Theorem 1. A bipartite graph $G = (U, V, E)$ has a perfect matching if and only if for every subset $U' \subset U$, we have $|\Gamma(U')| \geqslant |U'|$.

Prove this theorem.

2d. Theorem 2. Every k-regular bipartite graph has a perfect matching. *Hint: for the proof you are allowed to use Theorem 1 even if you have not proved that correctly. Of course you are not allowed to use Theorem 3 below ;-)*

Prove this theorem.

2e. Theorem 3. Every k-regular bipartite graph has k edge-disjoint perfect matchings, i.e. k perfect matchings such that none of them has edges that also belong to another one. *Hint: for the proof you are allowed to use Theorem 2 even if you have not proved that correctly.*

Prove this theorem.

3. Consider the MAKESPAN scheduling problem.

MAKESPAN:
Instance: Given a set of n jobs and for each job j a processing time p_j, $j = 1, \ldots, n$ and m parallel identical machines. A feasible schedule of the jobs is an assignment of the jobs to the machines and an ordering of the jobs, such that each job is processed on only one machine and each machine processes at most one job at a time.
Objective: Find a feasible schedule with minimum makespan, i.e. the time the last job completes.

(**a**) Formulate MAKESPAN-DECISION, the decision version of MAKESPAN and show that it is NP-Complete if $m = 2$ (on 2 machines).
Hint: Use a reduction from PARTITION*. A correct set-up of the proof without the exact proof also earns you some points.*

(**b**) Use the result of (**a**) to prove that MAKESPAN-DECISION on 3 machines is NP-Complete.
Hint: Easiest is to introduce an artificial (long) job in the reduction. Also here, a correct set-up of the proof without the exact proof also earns you some points.

(**c**) Design a PTAS for the MAKESPAN problem on 3 machines. Include the running time analysis.
Hint: The way the PTAS is constructed for the 2-machine version is useful here.

4. Consider the HAMILTONIAN WALK problem.

HAMILTONIAN WALK:
Instance: Given a (unweighted) graph $G = (V, E)$. A Hamiltonian Cycle in the graph is a closed walk that visits every vertex of the graph *at least* once.
Objective: Find a Hamiltonian Cycle with minimum total length (number of edges/vertices on the closed walk).

(**a**) Prove that this problem is *NP-hard*. *Hint: Easy reduction from Hamiltonian Circuit*

(**b**) Design a polynomial time approximation algorithm A that has an approximation ratio

$$\frac{Z^A(I)}{Z^{OPT}(I)} \leqslant \frac{3}{2}$$

for every instance I of the problem, and prove this ratio. *Hint: Remember metric TSP*