Exam (resit) Combinatorial Optimization, 12 December 2017, 12.00- 14.45

- It is not allowed to use any books, notes, or calculator. Just pen and paper.
- The table shows the maximum number of points per (sub-)question.

1a	1b	1c	1d	1e	2a	2b	3	4a	4b	5a	5b	5c	6a	6b	7a	7b	7c	$\sum$
1	1	1	2	2	2	3	5	2	3	3	2	2	2	2	2	2	3	40

---

1. Consider the following instance of the weighted set cover problem. The elements are $E = \{1,2,3,4,5\}$ and the subsets are $S_1 = \{1,2,3\}$, $S_2 = \{2,3,4\}$, $S_3 = \{3,4,5\}$, $S_4 = \{1,4,5\}$, and $S_5 = \{1,2,5\}$. The weights of the sets are $w_1 = 10$, $w_2 = 10$, $w_3 = 10$, $w_4 = 12$, and $w_5 = 8$.

(a) Write down the ILP for this set cover instance. What is the optimal value?

(b) Give a solution to the LP-relaxation with value strictly less than the optimal ILP value.

(c) Write down the dual of the LP-relaxation for this instance.

(d) Apply the primal-dual algorithm (from chapter 1) for this instance. Explain how you raise the dual variables step by step. Which sets are chosen in the final solution?

(e) Argue why in general this primal-dual algorithm always returns a feasible solution.

2. Consider the following minimization problem:

DEGREE BOUNDED SPANNING TREE:

Instance:	Graph $G = (V,E)$
Solution::	A spanning tree T of G
Value:	Maximum degree of T
Goal:	Find a solution with minimum value.

(a) It is well-known that the Hamiltonian path problem is *NP*-hard. Use this to show that the degree bounded spanning tree problem is *NP*-hard.

(b) Let $\alpha < 3/2$. Show that there is no polynomial time α-approximation algorithm for this problem (assuming that $P \neq NP$).

3. The list scheduling algorithm is known to be a 2-approximation algorithm for minimizing the length of the schedule ('makespan') on identical parallel machines. Assume now that all jobs are relatively small. Precisely, assume that for each job k, its processing time p_k satisfies

$$p_k \leq \frac{1}{3(m-1)} \sum_{j=1}^{n} p_j,$$

where m is the number of machines and n is the number of jobs. Prove that in this case, list scheduling is a $4/3$-approximation algorithm.

4. **(a)** Describe Christofides' algorithm for the metric TSP.

(b) Show that the approximation ratio of the algorithm is no more than $3/2$.

5. Consider the following minimization problem from Chapter 4.

PRIZE-COLLECTING STEINER TREE:

Instance:	$G = (V,E)$ and a cost c_e for every edge $e \in E$ and a penalty π_i for every vertex $i \in V$. Also given is a root $r \in V$.
Solution:	Tree T containing r. Let $V(T)$ be the vertices in T
Value:	$\sum_{e \in T} c_e + \sum_{i \in V - V(T)} \pi_i$.
Goal:	Find a solution of minimum cost.

(a) Chapter 4 gives an algorithm using LP-rounding. Give the ILP used.

(b) The ILP (and its LP-relaxatian) has an exponential number of constraints. In general, an LP with an exponential number of constraints can be solved in polynomial time if it has a so called separation oracle. Explain what a separation oracle is.

(c) The LP-relaxation of the ILP for this problem does have a separation oracle. Explain how.

6. Consider the following graph coloring problem.

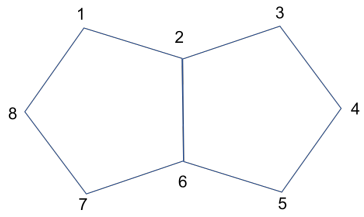
MAX. 3-COLORING:

Instance:	A Graph $G = (V,E)$
Solution:	A value (color) $x_v \in \{1,2,3\}$ for each vertex $v \in V$
Value:	The number of edges for which the two endpoints have different colors.
Goal:	Find a solution of maximum value.

(a) Give a randomized $2/3$-approximation for the problem. (That means, give the algorithm and prove that the expected value is at least $2/3$ times the optimal value).

(b) Describe how the algorithm can be derandomized.

7. **(a)** Give the vector program for computing a coloring of a 3-colorable graph (as given in Section 6.5). Either give the general formulation or give the formulation for the graph below.



(b) Show that the optimal value of the Vector Program is no more than $\cos(\frac{4\pi}{5})$ for the graph below.

(c) Assume we apply one iteration of the algorithm of Section 6.5 with $t = 2$ hyperplanes. That means, we take two hyperplanes at random and color each vertex with a color from $\{1,2,3,4\}$. Show that the probability that this gives a feasible 4-coloring is at least $16/25$.