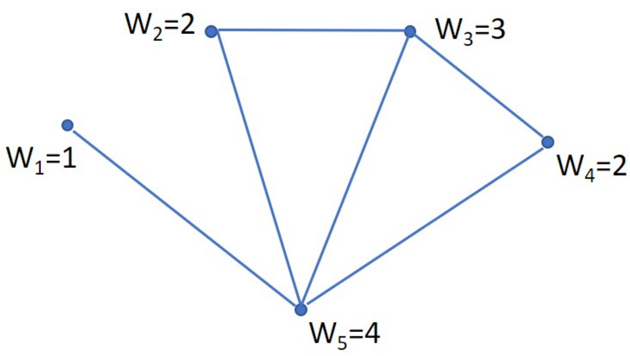
Exam Combinatorial Optimization - Resit , 15 December 2021.

- It is **not** allowed to use any books, notes, or calculator. Just pen and paper.
- You may also obtain points for partial answers.
- Clearly indicate where your final answer is in case you make some computations.

1a	1b	1c	2a	2b	3a	3b	4a	4b	5	6a	6b	7a	7b	7c	Σ
2	3	3	2	3	4	3	2	3	7	3	3	2	2	3	45

Question 1 (8 points)
The figure shows in instance of the weighted vertex cover problem.



(a) (2pt.) What is the optimal value?

(b) (3pt.) Write down the Vertex Cover ILP for this instance. Give an solution to the LP-relaxation of value 6.

(c) (3pt.) Write down the dual of the LP. Give an solution to the dual of value 6.

Question 2 (5 points)
In the maximum k-cut problem, we are given an undirected graph $G = (V,E)$, and non-negative weights $w_{ij} \geq 0$ for all $(i, j) \in E$. The goal is to partition the vertex set V into k parts $V_1,\dots,V_k$ so as to maximize the weight of all edges whose endpoints are in different parts.

(a) (2pt.) Describe a *randomized* $(1-1/k)$-approximation algorithm for the maximum k-cut problem. (Proof in part (b)).

(b) (3pt.) Prove that the expected ratio is $(1-1/k)$.

Question 3 (7 points)
A *triangle* in a graph $G = (V, E)$ is a clique of size 3, i.e, a set of vertices $v, w, z \in V$ such that $(v, w), (w, z)(v, z) \in E$.

TRIANGLE HITTING SET:

Instance:	A Graph $G = (V, E)$
Solution:	A set $S \subseteq V$ such that each triangle has at least TWO vertices in S.
Value:	The number of vertices in S
Goal:	Find a solution of minimum cost.

(a) (4pt.) Give an ILP formulation for the Triangle Hitting Set problem. Use variables $x_j \in \{0, 1\}$ for all $j \in V$.

(b) (3pt.) Describe a 2-approximation algorithm for the Triangle Hitting Set problem using LP-rounding. (Give the algorithm and give a proof for the ratio of 2.)

Question 4 (5 points)

$$C_1 = x_1 \vee \overline{x_2}, \qquad C_2 = \overline{x_1} \vee \overline{x_3} \qquad C_3 = x_2 \vee x_3, \qquad C_4 = x_1$$

(a) (2pt.) Shown is an instance of the (unweighted) maximum satisfiability problem. What is the expected number of satisfied clauses when you assign each variable independently the value 'True' with probability 0.5? Show your computation.

(b) (3pt.) Now suppose that you apply the method of conditional expectations to derandomize this approach. Assume that you go through the variables in the order x_1, x_2, x_3. Then what assignment do you get? Show your computation.

Question 5 (7 points)
A special case of the TSP-problem is the one in which all edge costs (distances) are either one or two: $c_{ij} \in \{1, 2\}$ for all pairs of points i, j. In other words, the instance is a *complete* graph and each edge has cost either 1 or 2. Here is a possible approximation algorithm:

Algorithm:
Step 1. Find a minimum cost 2-matching in the graph.
Step 2. Turn the 2-matching into a TSP-tour by deleting some edges and adding some edges.

Step 1 can be done in polynomial time. (Remember that a 2-matching is a subset of the edges such that each vertex is adjacent to exactly two of them and the cost of the 2-matching is defined as the sum of the edge costs.)

Step 2 can be done in such a way that the resulting TSP-tour is a $4/3$-approximation.

Describe how step 2 is done and prove that this gives indeed a $4/3$-approximation. (Make a picture.)

Question 6 (6 points)
For this question, we first recap some theory of the Prize-collecting Steiner Tree problem.

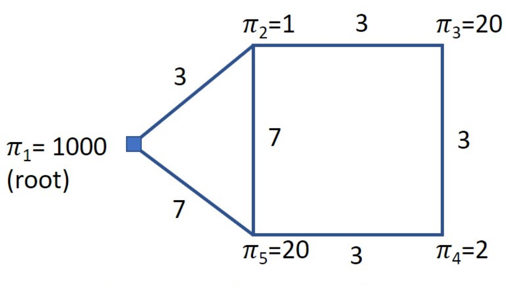
PRIZE-COLLECTING STEINER TREE:

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.

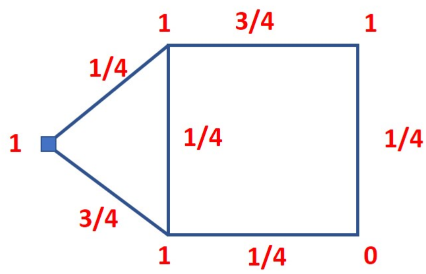
An ILP for the problem is as follows.

$$
\begin{array}{lll}
\text{(ILP)} \quad \min & Z = \sum\limits_{e\in E} c_e x_e + \sum\limits_{i\in V} \pi_i(1-y_i) & \\
s.t. & \sum\limits_{e\in\delta(S)} x_e \geqslant y_i, & \text{for all } i,S \text{ with } i \in S \subseteq V-r, \\
& y_r = 1, & \\
& x_e \in \{0,1\}, & \text{for all } e \in E, \\
& y_i \in \{0,1\}, & \text{for all } i \in V.
\end{array}
$$

In the LP-relaxation, the binary constraints are replaced by $0 \le x_e \le 1$ and $0 \le y_i \le 1$.



Instance with edge- and penalty costs



x- and y-values

(a) (3pt.) What is the optimal value for the instance shown above? (Left figure only. The figure on the right is for (b).)

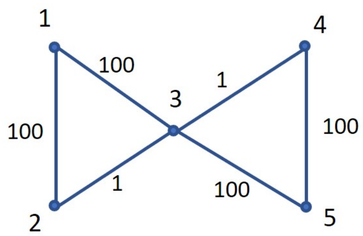
(b) (3pt.) Is the solution shown on the right a feasible solution to the LP-relaxation? Explain your answer.

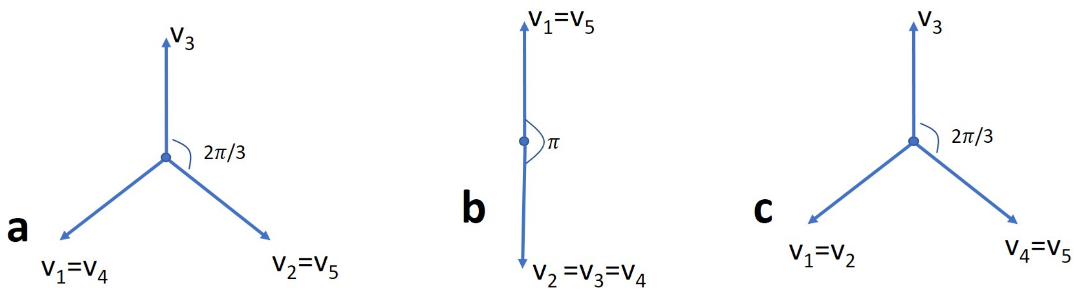
Question 7 (7 points)

Shown below are the Quadratic Program and the Vector Program relaxation for the maximum weighted cut problem, where an instance is an edge-weighted graph $G = (V, E)$ on n vertices.

$$\text{(QP)} \quad \max \quad \frac{1}{2} \sum_{(i,j)\in E} (1 - y_i y_j) w_{ij}$$
$$s.t. \quad y_i \in \{-1, 1\} \qquad i = 1, \ldots, n.$$

$$\text{(VP)} \quad \max \quad \frac{1}{2} \sum_{(i,j)\in E} (1 - v_i \cdot v_j) w_{ij}$$
$$s.t. \quad v_i \cdot v_i = 1,\ v_i \in \mathbb{R}^n \qquad i = 1, \ldots, n.$$





(a) (2pt.) What is the value of the maximum weighted cut in the graph above? (See top figure.)

(b) (2pt.) Below the graph are 3 feasible solutions to the VP for this instance. Rank the solutions from largest to smallest VP-value. (You do not need to give the value. Just answer 'a, b, c' for example.)

(c) (3pt.) Remember that the max-cut algorithm of Goemans and Williamson (of Chapter 6) was defined as follows: Solve the VP and then apply randomized rounding. Now suppose that instead of the *optimal* VP-solution you take each of the 3 solutions above. For each of the 3 solutions give the the expected weight of the cut.