

EndMidTerm January 8, 2016.

- Duration 2h. One Cheat Sheet allowed (A4, manuscript, double sided). -

- **Exercise 1 - Simplex.** Solve the following linear program by the two-phase simplex algorithm (and only by this method).

$$\begin{array}{llll} \text{Maximize} & 3x_1 & +x_2 & \\ \text{Subject to} & x_1 & -x_2 & \leq -1 \\ & -x_1 & -x_2 & \leq -3 \\ & 2x_1 & +x_2 & \leq 4 \\ & x_1, & x_2 & \geq 0 \end{array}$$

Write the dual of (P) and solve it using your previous simplex.

- **Exercise 2 - Branch and bound.** To compute an optimal integer solution of some maximization (standard) linear program (P), one can apply the following branching algorithm:

First of all, solve (P) fractionally, and get an optimal solution $x = (x_1, \dots, x_n)$. Then, unless x is already integer, consider a non-integer x_i , chosen as non integer as possible (i.e. minimize $|1/2 - x_i + \lfloor x_i \rfloor|$). Now create two new linear programs: (P₁) obtained from (P) by adding the constraint $x_i \leq \lfloor x_i \rfloor$ and (P₂) by adding the constraint $x_i \geq \lceil x_i \rceil$...

- a. Finish the description of the algorithm and show that it provides an optimal integer solution. Provide a decent heuristic.
- b. Apply your algorithm to the linear program (here using simplex is not mandatory):

$$\begin{array}{llll} \text{Maximize} & 2x_1 & +3x_2 & \\ \text{Subject to} & x_1 & +2x_2 & \leq 3 \\ & 6x_1 & +8x_2 & \leq 15 \\ & x_1, & x_2 & \geq 0 \end{array}$$

- **Exercise 3 - No comment.** We assume that x^* is an optimal solution of some linear program $\text{maximize } \{c^T x \mid Ax \leq b, x \geq 0\}$. Suppose moreover that A and b decompose as

$$A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

where $A_1 x = b_1$ and $A_2 x < b_2$ (i.e. $<$ on each row). Show that x^* is also an optimal solution to the linear program $\text{maximize } \{c^T x \mid A_1 x \leq b_1, x \geq 0\}$.

- **Exercise 4 - TU matrices.** Let A be a totally unimodular matrix.

- a. Show that the set of columns of A can be partitioned into *blue* columns and *red* columns in such a way that on each row, the sum of blue values and the sum of red values differ at most by one. Hint: propose a fractional relaxation of the problem, and use TU property.
- b. Illustrate this property on some classical TU matrices (and their transpose!). For each example provided, propose a direct ad-hoc proof.

- Exercise 5 - Multiflow. Let $D = (V, A)$ be a directed graph and $c : A \rightarrow \mathbb{R}^+$ be a capacity function on the arcs of D . We are given k requests $R_i = (s_i, t_i, d_i)$ in which s_i, t_i are vertices of V and d_i is the demand of request R_i . Moreover, every arc a has a cost $s(a)$. The multiflow problem asks for the minimum cost satisfaction of all requests R_i as disjoint (s_i, t_i) -flows. Fractional solutions are allowed.

- a. Express the multiflow problem as a linear program.
- b. Assume that you are given an instance of multiflow together with a length function on the arcs $l : A \rightarrow \mathbb{R}^+$ satisfying:

$$\sum_{i=1}^k d_i l(s_i, t_i) > \sum_{a \in A} c(a) l(a)$$

where $l(s_i, t_i)$ is the length of a shortest path (with respect to length l) from s_i to t_i . Show then that a solution to multiflow does not exist.

- c. Show that the integer multiflow problem is NP-hard using the fact that deciding the existence of a circuit containing two prescribed vertices in a directed graph G is NP-complete.
- d. How to justify that a (fractional) multiflow instance has no solution?