INTRO TO APPROXIMATION, CLASS 3

linear programming in a nutshell

Příklad první *Introductory example:* A polar expedition tries to pack the most caloric food which can be bought for just 5,000 CZK. Design a linear program which finds the most caloric combination if they can choose from the the following food.

Food	CZK per 100 g	Kcal na 100 g
Cheese	20	320
Bread	4	120
Dry sausage "Vysočina"	16	500
Pork	19	240

Exercise two Formulate a linear program for finding an s-t flow in a network of amount d and of minimal cost. Network is given as a directed graph $G = (V, E)$ with edge capacities c (the edge from u to v has capacity $c_{uv} \geq 0$), edge costs w (sending x_e of flow through an edge e costs $w_e \cdot x_e$), source s and target t. Additionally, you are given $d \geq 0$ which is the required amount of the s-t flow.

Next we deal with *integer linear programs* (ILP), which are like linear programs, but (some) variables must have integer values only.

Exercise three *Warming up with a knapsack!*

Formulate the knapsack problem using ILP: There are n items with weights w_i and values v_i, we have a knapsack with weight capacity W and we are packing items into the knapsack to maximize the total value of packed items.

Exercise four In the Vertex Cover problem we are looking for a set U of vertices of the smallest possible size so that each edge of the graph is covered, that is, at least one of its two endpoints is in U.

a) Formulate the problem using integer programming.
b) We often transform an ILP into a linear program, called the *relaxation*, by dropping the condition that the variables should be integer. Then we solve the relaxation (in polynomial time). However, we are interested in an integer solution. Find an easy way how to get a *feasible* integer solution from an optimal solution of the relation such that the integer solution is a 2-approximation.

Exercise five Formulate an ILP for 3-Partition: We have a set of $3n$ items with values $v_1, \ldots, v_{3n}$ and we want to find whether we can split them into n triples so that each triplet has the same sum S.

Exercise six Jack got the following exercise during the class:

Formulate an ILP which finds the shortest Hamiltonian circle, that is, for a graph $G = (V, E, d)$ with edge lengths $d(e) \geq 0$ we want to find the shortest cycle which visits each vertex exactly once.

Jack suggests the following solution:

„For each edge uv we have a variable $x_{uv} \in \{0, 1\}$, objective function is $\min \sum_{uv \in E} d(uv) x_{uv}$ and for each vertex u we have a constraint $\sum_{v | uv \in E} x_{uv} = 2$.“

Is this solution correct?

If you are already familiar with linear programming, you can try to solve the following exercises in addition to the exercises above (I also recommend solving the exercise about Vertex Cover).

Formulate an ILP for the following problems. When you are done, you can try to find an instance where an integer optimal solution has a high value, while there is a real solution of the relaxation with a substantially smaller value. (You can also try to dualize the relaxations.)

Exercise seven   Chromatic Number of a Graph: For a given undirected graph find the smallest possible number of colors such that each vertex is assigned a color and no two neighboring vertices have the same color.

Exercise eight   Scheduling Jobs with Release Times and Deadlines: We have one machines and a set of n jobs. Each job j has a release time r_j, a deadline d_j, a processing time p_j and a value v_j If j is scheduled, it must start at r_j or later and it must be finished till d_j. The goal is to maximize the total value of scheduled jobs (of course, no two jobs may run at the same time) All release times, deadlines and processing times are integers.