Métodos de Apoio à Decisão
Assignment 2: An assignment problem

João Pedro PEDROSO

DCC - FCUP, April 2022

In this assignment there will be some questions based on the following exercises. In the assignment's class there will be a set of questions in Codex*, with the computers set up as in previous classes. The AMPL book and the classes' slides will be available for consulting.*

Note 1: *You will be able to use `glpsol` and/or the commercial software AMPL (`https://ampl.com`); a version with a license for this course is available in `https://www.dcc.fc.up.pt/~jpp/AMPL`. A well-known solver for dealing with integer optimization problems is `gurobi`.*

Note 2: *You will be given scratch paper. Please do not use any other materials or electronic devices during the class.*

Note 3: *Each pair of students cannot do more than one assignment together.*

Exercise 1

After completing your degree, you and some colleagues decided to start a company UBRbytes, with the aim of connecting software companies to programmers like you. Your platform will announce tasks that software companies want to be completed in the following week, and programmers will announce their availability for programming components required by these tasks. The task is to match programmers, based on their skills, to the skills required for completing each task. The difference between what the software companies pay for tasks completed and what the programmers charge will be your profit.

For instance, you may have available 3 programmers, able to code in C, Python, and Web applications as indicated in the following table, where 1 means compatible and "."/0 means incompatible.

	p1	p2	p3
c	1	.	1
py	1	1	.
web	1	1	1

Notice that this table can be read as a parameter in AMPL data attributing a zero to ".", e.g., if it is declared as:

```
param a {SKILLS, TASKS} default 0;
```

Next week, each of them is available the following number of hours, at the cost per hour:

	hours	cost/hour
p1	4	5
p2	10	5
p3	7	5

There are 5 tasks proposed by software companies for next week, at the following price:

t1	250
t2	150
t3	200
t4	100
t5	250

The number of hours required per programming skill are:

	c	py	web
t1	5	.	2
t2	.	3	.
t3	.	6	3
t4	3	3	3
t5	2	3	2

These tasks cannot be divided: a programmer can only be assigned to a task if she can complete all its subtasks. For example, programmer p1 cannot use her 4 hours for C programming in task t1, as the task requires 5 hours of C programming plus 2 hours of web programming. She could, however, do task 2, which only requires Python programming; then, she would have 1 hour remaining, which could not be used in any other task, as they all require more than 1 hour.

The optimal solution in this case would be p1 doing t2, p2 doing task t3 (he can both program Python and web, and has available more than the 9 hours required), and p3 doing task t1. Tasks t4, t5 are not covered. The optimal profit is 505 euros.

Exercise 1.1: Prepare a model, independent of the actual data (so that it is easier to solve with different data), for solving this problem.

Exercise 2

As a way of improving customer satisfaction, you decided to keep track of the evaluation given by the customers to each of the programmers; 5 is the best evaluation, 1 is the worst, 0 if no information is available. For example, there could be

	p1	p2	p3
c	5	.	5
py	5	3	.
web	1	5	1

Exercise 2.1: Solve the problem for the case where your company wishes to maximize the evaluation of programmers assigned to tasks that will be completed, summed for all the subtasks and all the programmers. Consider the same constraints as in Exercise 1, and a constraint imposing that the profit must not be worse than when maximizing profit.

Exercise 3

You have realized that there is demand for tasks that require more programming hours than any of the programmers provides alone. Though ideally all tasks should be made by the same person, you may propose to some programmers to form a team in the following week, and divide the work required to complete a task. Your application must decide how many hours each of them works, on each subtask, as their payment will be in that proportion.

Exercise 3.1: Solve the problem of maximizing profit in this situation.

Exercise 3.2: Solve the problem of minimizing the number of persons in the team with most elements (the largest team) subject to a profit at least as large as determined Exercise 3.1.