

Programming Assignment 3

Use the PuLP library <https://pypi.org/project/PuLP/> to solve the following problems. Documentation to PuLP can be found here: <https://coin-or.github.io/pulp/main/index.html>

Bakery problem (20 points)

Consider a small bakery with a single oven. It needs to schedule baking of n pastries, each of them having three requirements:

- Time when the preparations are done and pastry is ready for baking.
- Time needed for baking, i.e., for how long should it remain in the oven.
- Deadline: time when the customer comes to pick up the pastry.

At each moment, only one kind of pastry can be present in the oven.

Use an ILP to find a shortest baking schedule. Schedule, in this context, is a set of starting times s_1, \dots, s_n denoting when should each pastry be put into oven. Note: these times need not be integral. However, integral variables will be useful to enforce that the periods when two different kinds of pastries are in the oven do not overlap.

Let us denote e_1, \dots, e_n the ending times of baking of each of the pastries, i.e., $e_i = s_i + \text{baking time of pastry } i$. We need to make sure that for each two pastries i, j , one of the following needs to be true: $e_i \leq s_j$ or $e_j \leq s_i$. Obviously, they cannot hold at the same time and it depends on the precedence between i and j which one is true. Since we do not know the precedence in advance, which of these constraints should we include in the LP?

Big-M method

This name usually refers to an alternative way how to start the simplex method without knowledge of the initial basic feasible solution. We did not cover this in the class and I do not go into details of this here either. But the other meaning of Big-M is a method for switching some of the constraints on/off depending on the value of some binary variable.

Imagine, we have variable x which should be bounded by 10 if and only if some binary variable z is set to zero. Also, assume that there is no reason to increase x beyond some large number M (e.g., because we are minimizing over x , or we know that no feasible solution can have $x > M$ for some other reasons). Then, we can write $x \leq 10 + M \cdot z$: if z is 0, this switches the constraint ON. If $z=1$, this constraint evaluates to $x \leq 10 + M$ which, by choice of M is satisfied by any reasonable solution to our LP and this effectively switches the constraint OFF. Usually, due to possible numerical issues, it is recommended to use M as small as possible. You can check the following blog for more discussion of big-M: <https://orinanobworld.blogspot.com/2011/07/perils-of-big-m.html>.

You may check that there is a suitable choice of M in our problem and use this approach in your solution.

Input

Text file containing a single line for each kind of pastry consisting of four numbers (integers) separated by spaces:

ID PRE DLN BAK

ID denotes the numerical ID of the pastry, PRE the time since midnight since when the pastry is ready for baking, BAK is the time it needs to spend in the oven, and DLN is the deadline when the pastry needs to be surely finished.

All times are in seconds.

Output

Dictionary containing starting time of each pastry. For pastry with ID i :

`retval['s_i']` = starting time of baking of pastry with ID i

Performance

In Mixed ILP, the performance becomes quite an issue and it matters how do you specify your program. In the template file, it is shown how to setup your LP solver to run in parallel in order to use the hardware of gradescope more efficiently. Since there are limits in gradescope on the length of the computation done during grading, your code when run in gradescope should finish within 10 minutes. Please try it before submitting!

Bakery problem visualization (10 points)

Use library matplotlib to visualize your solution suitably. I leave to your creativity how to do it, but it should be clear what are the moments when oven needs to be open, what pastry goes out and what should be put in. There are many other things to visualize: expected arrivals of customers and times when each pastry is ready, critical preparations (which pastry needs special care to be prepared on time, otherwise it would delay the whole schedule, etc). The visualization should be understandable to a non-expert, e.g. a baker operating the oven. Therefore, the main criterion for evaluation of this will be clarity and information it provides.

Output:

Your program should produce a picture in PNG format and record it to the file
./visualization.png

Submission

Submit both `main.py` containing your code and `visualization.png` produced on your computer.