Instructions for AI & optimization: lab 5

Isep Paris, 6th April 2022

I. The main instructions

Complete the Jupyter notebook **lab05.ipynb**, and send it back on Moodle by **Wednesday, 6th April 2022 12:15.**
You may work alone, or in a **team made of two students maximum. Penalties will be applied for students who do not respect this rule.**
Do not forget to write the name of ALL the members of your team on the associated cell on your modified notebook. A student associated with no notebook will have 0 for this lab.

If you are absent during the lab 5 session, and do not send a justification to edith.chan@isep.fr, you will be sanctioned by a mark of 0 for this lab, even you magically send a report on Moodle by noon…
Please also note that penalties will also be applied on any report mentioning the name of an absent student.

The deposit will be closed after the deadline. No delay will be tolerated. If you encounter any technical problem, you may send your work at helene.urien@isep.fr by email. If you are absent during the lab 5 session, your work will still not be evaluated, even if you send it by email.

Additional instructions :

- Run your Jupyter notebook before sending it back (kernel → Restart & Run all).
- Check that there is no error message.
- Do not write any space or special character in the name of your file
- Ensure that each line of code does is not (roughly) composed of more than 80 characters

II. Libraries to load

If not done yet, you need to install the following libraries:

- **Pil (question 1)**
- **Scikit-learn (question 2)**
- Scikit-image (*skeletonize* function after question 3, strongly recommended)
- **Scipy (question 5)**
- Scikit-network (from question 6, version 0.18.0 recommended, not mandatory)

Note : Scikit-network is a Scipy-based library for graph processing. You may use one of the two libraries to perform shortest path methods. Using Scikit-network, you can directly find the shortest path between two given nodes. If you choose Scipy, you can specify the starting node, but not the ending point : Scipy computes all the possible shortest paths starting this input point. However, the shortest path to a specific node can be easily found using the output **predecessors array.**