GROUPE ISI	Institut Supérieur Informatique	
	TP de Programmation Linéaire	Prof. M. Sy Periode: 2019 – 2020
	Classe(s): Licence 2 *RI* & GL	Durée: 2H

Pré-requis

Pour bien suivre ce TP il faut au préalable avoir :

1. une version de python3 installer `https://www.python.org/downloads/`
2. Ensuite vous taper dans votre CMD:

```
pip install notebook
```

   pour installation de *jupyter notebook*, vous pouvez consultez le site aussi `https://jupyter.org/install`

3. Enfin vous devez installer ces packages toujours avec pip qui est un gestionnaire de paquets utilisé pour installer et gérer des paquets écrits en Python.En tapant dans votre CMD

```
pip install numpy
pip install scipy
pip install matplotlib
```

Charger le module : *scipy.optimize*

SciPy optimize provides functions for minimizing (or maximizing) objective functions, possibly subject to constraints. It includes solvers for nonlinear problems (with support for both local and global optimization algorithms), linear programing, constrained and nonlinear least-squares, root finding, and curve fitting.

Linear programming : importer la fonction *linprog*

linprog(c[, A_ub, b_ub, A_eq, b_eq, bounds, ...])	Linear programming: minimize a linear objective function subject to linear equality and inequality constraints.

The linprog function supports the following methods:

- linprog(method='simplex')
- linprog(method='interior-point')
- linprog(method='revised simplex')
- linprog(method='highs-ipm')
- linprog(method='highs-ds')
- linprog(method='highs')

The simplex, interior-point, and revised simplex methods support callback functions, such as:

linprog_verbose_callback(res)	A sample callback function demonstrating the linprog callback interface.

Linear programming : Comment implémenter *linprog*?

scipy.optimize.linprog(*c, A_ub=None, b_ub=None, A_eq=None, b_eq=None, bounds=None, method='interior-point', callback=None, options=None, x0=None*) [source]

Linear programming: minimize a linear objective function subject to linear equality and inequality constraints.

Linear programming solves problems of the following form:

$$\begin{aligned} \min_x \ & c^T x \\ \text{such that } & A_{ub} x \leq b_{ub}, \\ & A_{eq} x = b_{eq}, \\ & l \leq x \leq u, \end{aligned}$$

where x is a vector of decision variables; c, b_{ub}, b_{eq}, l, and u are vectors; and A_{ub} and A_{eq} are matrices.

Je vous recommande de consulter ce site pour plus de détail: `https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.linprog.html`