

TP 1.5

Pour aller plus loin que le TP1. En est la suite directe.

Corrélation aux complexités théoriques

Le rapport entre le temps mis pour effectuer un algorithme et sa complexité théorique doit être une droite. Utilisez `numpy` et `polyfit` pour calculer la droite de régression linéaire de ce rapport pour l'algorithme naïf et l'algorithme rapide.

```
import numpy

x = list(range(10))
y = [5 * i + 10 for i in x] # y = 5x + 10

# régression linéaire
a, b = numpy.polyfit(x, y, 1)
print(a, b) # a = 5, b = 10
```

En déduire le coefficient de corrélation linéaire (on pourra utiliser la méthode `corrcoef` de `numpy`).
Conclusion ?

Le débugeur

En plus des tests, pouvoir exécuter ligne à ligne un algorithme est un outil puissant pour corriger ou comprendre un algorithme. Cet outil s'appelle un debugger et Python et pycharm en possèdent un.

Essayez d'exécuter ligne à ligne l'algorithme rapide avec le debugger de pycharm en suivant ce [manuel](#).

From:

<https://wiki.centrale-med.fr/informatique/> - WiKi informatique

Permanent link:

<https://wiki.centrale-med.fr/informatique/restricted:alg-1:tp1.5>

Last update: **2016/11/24 11:13**

