

## Fiche méthode n° 5

## LatisPro : import des données sous python

On peut récupérer les données des fichiers CSV sous python pour les exploiter.

Temps	EA0	Temps	EA1
0	-8,27815887611359	0	-4,95042419806123
1E-05	-8,258200683631	1E-05	-4,76622699201107
2E-05	-8,21828429866582	2E-05	-4,57705148309469
3E-05	-8,17836791370064	3E-05	-4,39783257991076
4E-05	-8,11849333625287	4E-05	-4,20865707099438
5E-05	-8,04365011444315	5E-05	-4,02445986494422
6E-05	-7,9538382482715	6E-05	-3,83528435602784
7E-05	-7,85404728585854	7E-05	-3,65108714997768
8E-05	-7,73928767908365	8E-05	-3,46191164106131
9E-05	-7,60456987982616	9E-05	-3,27273613214493
0,0001	-7,47484162868932	0,0001	-3,09351722896099
0,00011	-7,32515518506989	0,00011	-2,90932002291083

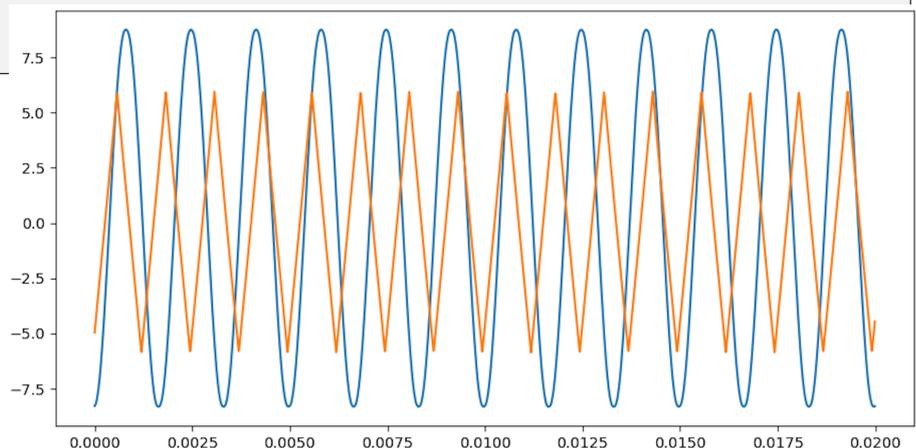
- La première ligne contient le descriptif des données pour chaque colonne.
- Les valeurs des instants de mesure sont dans la 1<sup>re</sup> colonne ainsi que dans la 2<sup>e</sup> colonne.
- Les valeurs des tensions mesurées sont dans la 2<sup>e</sup> colonne pour la voie EA0 et dans la 1<sup>re</sup> colonne pour la voie EA1.
- La virgule est utilisée comme symbole décimal.
- On peut récupérer les instants de mesure indépendamment sur la voie EA0 ou sur la voie EA1, ils sont identiques.

Les données sont stockées dans le fichier deux-voies.csv placé dans le répertoire de travail.

```

1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 fichier = open("deux-voies.csv","r")
5 donnees_brutes = fichier.readlines()
6 fichier.close()
7
8 donnees = []
9
10 for i in range(len(donnees_brutes)):
11     donnees_brutes[i] = donnees_brutes[i][:-1]
12     donnees.append(donnees_brutes[i].split(";"))
13
14 temps,EA0,EA1 = [],[],[]
15
16 for i in range(1,len(donnees_brutes)):
17     temps.append(float(donnees[i][0].replace(",",".")))
18     EA0.append(float(donnees[i][1].replace(",",".")))
19     EA1.append(float(donnees[i][3].replace(",",".")))
20
21 plt.plot(temps,EA0)
22 plt.plot(temps,EA1)
23 plt.show()

```



## Explication du code

**lignes 4 à 6 :** lecture des données brutes. On obtient une liste dont chaque élément est une chaîne de caractères constituée de la concaténation des caractères des 4 colonnes de la ligne de l'enregistrement. Par exemple `donnee_brutes[0]` retourne `'Temps;EA0;Temps;EA1'`  
`donnee_brutes[0]` retourne `'0;-8,27815887611359;0;-4,95042419806123'`.  
`len(donnees_brutes)` retourne le nombre total de lignes de l'enregistrement.

**lignes 10 à 12 :** on met en forme les données de chaque ligne.

- Chaque ligne se termine par un caractère de retour à la ligne. On supprime donc le dernier caractère de chaque élément (ligne 11).
- On construit la liste `donnees` dont chaque élément est une liste de 4 éléments, où un élément est la chaîne de caractères constituant chaque colonne de l'enregistrement (on scinde les lignes en éléments séparés par des points-virgules).  
`donnees[0]` retourne la liste `['Temps', 'EA0', 'Temps', 'EA1']`  
`donnees[1]` retourne la liste `['0', '-8,27815887611359', '0', '-4,95042419806123']`
- On a pris la première colonne pour extraire le temps (la 3<sup>e</sup> colonne est identique).

**lignes 16 à 19 :** on construit une liste par grandeur mesurées. Il faut convertir chaque élément, qui est une chaîne de caractères, en valeur numérique (flottant), en remplaçant la virgule par un point.