

Compter la nombre d'occurrences d'une lettre n'est pas difficile en soi, le problème vient du codage des caractères par la machine ! Longtemps la norme ASCII (American Standard Code for Information Interchange) a dominé, mais chaque pays a utilisé son propre encodage de caractères... dans tous les cas on ne pouvait accéder *que* à 128 caractères (pas forcément imprimables).

La norme Utf-8 (Universal Character Set Transformation Format - 8 bits) permet de coder des milliers de caractères.

Dans l'idée de « soit efficaces pour les élèves », je pense que l'exercice qui consiste à obtenir la fréquence d'apparition des lettres d'un texte doit se faire sur un texte préparé (pas d'accent, ni de signes de ponctuation ou autres symboles particuliers), tout en minuscule...)

Comprendre le programme suivant, le tester avec les différents textes préparés sur <http://frederic.leon77.free.fr/> (puis formations > bestiaire).

```
1  # -*- coding:utf8 -*-
2  # python 3
3  #
4  # problème de l'encodage : ASCII / Unicode
5  # https://python.developpez.com/cours/apprendre-python3/?
   ↪ page=page_12#L12-A-11-B
6  # dans ce programme, les fichiers sont préparés grâce à :
7  # https://www.dcode.fr/supprimer-accent-diacritiques
8  #
9  # pour les graphiques
10 # http://www.python-simple.com/python-matplotlib
11
12 from matplotlib import pyplot as plt
13
14 nb_occur = [0]*26 #tableau pour compter les occurrences de chq
   ↪ lettre
15 nb_lettres = 0
16
17 # préparation du fichier texte
```

Fréquence des lettres



```
18 with open('perec1_decode.txt', 'r') as mon_fic:
19     texte = mon_fic.readlines()
20
21 for ligne in texte:
22     ligne = ligne.lower() #met tous les codes caractères en
    ↪ minuscule
23     for c in ligne:
24         if 'a' <= c <= 'z':
25             nb_occure[ord(c) - 97] += 1
26             nb_lettres += 1
27 frequence = [nb_occure[k] / nb_lettres for k in range(26)]
28
29 for ind, freq in enumerate(frequence):
30     print(chr(ind + 97) + " --> ",freq)
31
32 plt.title("fréquence d'apparition des lettres")
33 plt.bar(range(26), frequence, width=.8, edgecolor = 'red', \
34         linewidth = .5, color='#8686F3')
35 plt.xticks([x + 0.5 for x in range(26)], [chr(c) for c in
    ↪ range(97,123)],\
36         rotation = 60)
37 plt.show()
```
