

Algorithmes de tri

1. Programmer une fonction `tri_naif(l)` qui prend en argument une liste et renvoie une liste contenant les même éléments rangés dans l'ordre croissant. On ne cherchera pas à optimiser la complexité de la fonction.
2. Programmer une fonction `fusion(l1, l2)` qui prend en argument deux listes triées, et renvoie une liste triée qui est l'union des deux listes. On cherchera à réaliser cette opération en complexité linéaire.
3. En utilisant la fonction `fusion`, écrire une fonction `tri_fusion(l)`, qui prend en argument une liste et renvoie une liste contenant les même éléments rangés dans l'ordre croissant. La fonction procédera de manière récursive en découpant la liste en deux, la triant chaque moitié, puis en fusionnant les deux listes ainsi triées. Quel est la complexité de cet algorithme ?
4. Charger le fichier `popMen.txt` ou `popWomen.txt`. Pour charger un fichier, on pourra utiliser la fonction `open()`, et la clause `with as` dont la syntaxe est

```
with open('chemin/vers/popWomen.txt', 'r') as file:  
    # instruction(s)
```

Où `'chemin/vers/popWomen.txt'` doit être remplacé par l'adresse du fichier dans votre ordinateur. Le fichier est alors accessible via la variable `file`.

5. convertir le contenu du fichier en une liste de nombres flottants. On pourra utiliser les méthodes ou procédures Python suivantes : `read`, `split`, `map` et `float`.

6. Comparer les temps d'exécution de la fonction `tri_naif` et `tri_fusion` sur la liste de nombre obtenue. Pour mesurer le temps d'exécution, on pourra utiliser le package `time`. Par exemple dans le code suivant :

```
import time
debut = time.time()
# instructions
fin = time.time()
duree = fin - debut
```

La variable `duree` contient la durée d'exécution en seconde des instructions qui se situent entre les deux appels de la fonction `time.time()`.

7. Comparer le temps d'exécution de la fonction `tri_fusion` avec la méthode `sort` fournie par Python (attention : l'appel de la méthode `l.sort()` modifie la liste `l` elle-même, et pas une copie).