

TD3 : Les listes

Xavier André & Romain Tavenard

1 Préambule

Comme pour le TD précédent, et comme pour tous les TD ce semestre, votre code pour ce TD devra se trouver dans un seul fichier, cette fois nommé `TD3_Listes.py` et stocké dans le même répertoire `M1_S1_Python`. La structure de ce fichier devra être la suivante :

```
1 # Section 1 : Imports de module
2
3 # Section 2 : Définition de fonctions
4
5 # Section 3 : Tests de fonctions définies et manipulations en mode "
    script"
```

2 Les listes

1. Écrire une fonction `print_sinus` qui affiche, pour chaque angle compris entre 0 et 360° (par pas de 1°), la mesure de l'angle en radian ($360^\circ = 2\pi$ radians) et son sinus.
2. Écrire une fonction `plusieurs_tables` qui prend en entrée une liste `l_bases` de bases et retourne les tables de multiplication pour ces bases sous la forme d'une liste de listes. Par exemple, si l'on fournit en entrée la liste `[2, 5]`, la fonction devra retourner la liste :

```
1 [
2     [2, 4, 6, 8, 10, 12, 14, 16, 18, 20],
3     [5, 10, 15, 20, 25, 30, 35, 40, 45, 50]
4 ]
```

3. Écrire une fonction qui prend en entrée trois entiers `borne_inf`, `borne_sup` et `puissance` et retourne la liste des entiers compris entre `borne_inf` et `borne_sup`, élevés à la puissance `puissance`.

4. **En utilisant la fonction codée à la question précédente**, écrire une fonction qui prend en entrée deux entiers `borne_inf` et `borne_sup` et retourne la liste des entiers compris entre `borne_inf` et `borne_sup`, élevés au carré.
5. Écrire une fonction qui prend en entrée une liste de chaînes de caractères `l_chaines` et retourne la liste des tailles des chaînes contenues dans la liste `l_chaines`.
6. Écrire une fonction `liste_triee_sans_doublon` qui prend en entrée une liste d'entiers quelconques `l_entree` et retourne une nouvelle version de cette liste sans doublon et triée. Vous ne devrez pas utiliser la fonction `set()` ici.
7. Écrire une fonction qui prend en entrée une chaîne de caractères et retourne le mot le plus long contenu dans cette chaîne.
8. Écrire une fonction qui prend en entrée une liste d'entiers et retourne une nouvelle version de cette liste qui ne conserve que les valeurs paires. Par exemple, si l'on fournit en entrée la liste `[2, 5, 7, 6]`, la fonction devra retourner la liste `[2, 6]`.

3 Pour aller plus loin : les listes en compréhension

1. Écrire une fonction qui prend en entrée une liste d'entiers `l_entree` et retourne la liste des valeurs de `l_entree` auxquels on aura ajouté la valeur 3. Le corps de votre fonction ne devra faire qu'une ligne.
2. Écrire une fonction qui prend en entrée une chaîne de la forme `"1.0 3.14 7 8.4 0.0"` et retourne la liste des réels contenus dans cette chaîne (ici `[1.0, 3.14, 7.0, 8.4, 0.0]`). Vous pourrez utiliser les fonctions `float()` et `split()` et le corps de votre fonction ne devra faire qu'une ligne.