TP 2.5

Pour aller plus loin que le TP2. En est la suite directe.

Nous allons implémenter deux tris rigolos.

Tri idiot

Implémentez le tri stupide et vérifiez sa complexité pour des listes de petite taille.

Le tri par base

Le tri par base (radix sort) est un tri utilisant l'écriture d'un nombre. Le but de cette partie est d'implémenter cet algorithme avec des nombres écrits en base 10.

Pour mener à bien ce tri il faut plusieurs choses.

Connaitre l'écriture d'un nombre en base 10

Pour cela on pourra transformer un nombre en chaine de caractère et accéder à ses chiffres. On pourra adapter le code suivant :

```
x = 1234
x_chaine = str(x)
print(len(x_chaine))
print(x_chaine[0])
print(x_chaine[-1])
```

Même nombre de chiffre pour chaque nombre

Pour que le tri se passe sans soucis, il faut s'assurer que chaque nombre soit écrit avec le même nombre de chiffre, quit à ajouter des "0" en début de nombre.

on pourra adapter le code suivant :

```
chaine_ajustee = "a".rjust(10, "b")
```

ranger les nombre par rapport à une position

On pourra utiliser des dictionnaire avec comme clé un chiffre allant de "0" à "9" et en valeur une liste. On pourra de la adapter code suivant :

```
rangement = dict()
for i in range(10):
  rangement[str(i)] = []

x = 1234
x_chaine = str(x)

position = -2
rangement[x_chaine[position]].append(x_chaine)
print(rangement)
```

A vous

Utilisez les parties précédentes pour créer le codage par base.