

# Chapitre 11 : Chaines - TD

## Exercice 1 : Comparaison de chaînes \*

En Python, on peut comparer 2 chaînes de caractères en utilisant les opérateurs <, >.... Il vous est demandé d'écrire une fonction `compareTo` qui prend deux chaînes `ch1` et `ch2` en entrée et renvoie

- +1 si `ch1` est après `ch2` dans l'ordre lexicographique
- -1 si `ch1` est avant `ch2`
- 0 sinon

sans utiliser les opérateurs entre chaînes précédemment cités !

## Exercice 2 : Recherche d'un caractère à l'intérieur d'une chaîne de caractères (`indexOf()`)\*

Un des traitements les plus courants est la recherche d'un (unique) caractère à l'intérieur d'une chaîne de caractères.

Ecrire une fonction `indexOfCar()` qui prend comme paramètres une chaîne de caractères `chaîne` et un (unique) caractère `car` et qui retourne l'index de la première occurrence de ce caractère dans la chaîne. Si `car` n'apparaît pas dans `chaîne` la fonction retournera -1. Cette fonction correspond à une fonction qui existe en Python et s'appelle `index()`. On vous demande toutefois de la définir vous même.

Par exemple, les instructions suivantes

```
In [ ]: chaine="il fait beau ici ; n'est ce pas ? oui c'est vrai"
        print(indexOfCar(chaine, 'e')) # conduira à l'affichage 9
        print(indexOfCar(chaine, 'x')) # conduira à l'affichage -1
```

## Exercice 3 : Recherche d'une sous-chaîne dans une chaîne de caractères (`find()`)\*\*

On peut généraliser le problème précédent en recherchant, non pas un unique caractère, mais une sous-chaîne dans une chaîne. Par exemple, dans la chaîne "il fait beau aujourd'hui", la sous-chaîne 'eau' apparaît à la position 9, mais on ne trouve pas la sous-chaîne 'uae'.

Ecrire une fonction `indexOfStr()` qui recherche la présence d'une sous-chaîne `sStr` dans la chaîne de caractères `chaîne`, toutes deux passées en paramètres. La fonction retournera la position de la première occurrence de `sStr` dans la chaîne de caractères `chaîne`; si celle-ci n'existe pas, la fonction retournera -1. Cette fonction correspond à une fonction qui existe en Python et s'appelle `find()`. On vous demande toutefois de la définir vous même.

Par exemple, les instructions suivantes

```
In [ ]: chaine="il fait beau ici, n'est ce pas ? oui c'est vrai"
        print(indexOfStr(chaine, 'bea')) # 8
        print(indexOfStr(chaine, 'aeb')) # -1
```

```
In [ ]: #Attention la syntaxe de find est différente
        chaine="il fait beau ici, n'est ce pas ? oui c'est vrai"
        print(chaine.find('bea'))
```

#### Exercice 4 : Comptage du nombre de voyelle(s) dans une chaîne de caractères\*

On rappelle que les voyelles sont les caractères {'a', 'e', 'i', 'o', 'u', 'y'}. Définir une fonction qui retourne le nombre de voyelle(s) (indifféremment majuscule(s) ou minuscule(s)) dans une chaîne de caractères et proposer un jeu de tests significatifs.

```
In [ ]: chaine="il fait beau ici;N'est ce pas:Oui c'est vrai"
        print("Le nombre de voyelles dans ' " + chaine + "' est : " + str(comptVoyelle(chaine)))
```

#### Exercice 5 : Remplacement d'un caractère par un autre (replace())\*

Ecrire une fonction changeCar(chaine, car1, car2) qui remplace tous les caractères car1 par des caractères car2 dans la chaîne de caractères chaine et retourne la chaîne résultante.

Cette fonction correspond à une fonction qui existe en Python et s'appelle find(). On vous demande toutefois de la définir vous même.

Par exemple, les instructions suivantes

```
In [ ]: chaine="il fait beau ici, n'est ce pas ? oui c'est vrai"
        print(changeCar(chaine, 'e','')) # il fait bau ici, n'st c pas ? oui c'st vrai
```

```
In [ ]: chaine="il fait beau ici, n'est ce pas ? oui c'est vrai"
        print(changeCar(chaine, 'e','')) # il fait bau ici, n'st c pas ? oui c'st vrai
```

#### Exercice 6 : Remplacement d'une sous-chaîne par une autre \*\*

Ecrire une fonction changeSsChaine(chaine,old,new) qui remplace tous les occurrence de la sous-chaîne old par la chaîne new dans la chaîne de caractères chaine et retourne la chaîne résultante.

Cette fonction correspond à une fonction qui existe en Python et s'appelle replace(). On vous demande toutefois de la définir (seule la fonction lenest autorisée).

Par exemple :

```
In [ ]: ch = "les hommes - tous les hommes - sont des animaux pensants"
        motAreplacer = 'hommes'
        motNouveau = 'êtres humains'
        print(changeSsChaine(ch,motAreplacer,motNouveau))
        # affichera : les êtres humains - tous les êtres humains - sont des animaux pensants
```