

# Algorithmique 4 : chaînes de caractères

---

## Exercice 1 :

Par défaut, en Python, l'instruction « `a=input()` » traite `a` comme une chaîne de caractères. Commenter chacune des instructions suivantes :

```
a=input("entrez un nombre entier (à huit chiffres par exemple) : ")
b=input("entrez un autre nombre entier (à six chiffres par exemple): ")
c=a+b
d=b*4
print(c)
print(d)
```

## Exercice 2 : manipulations usuelles

Commenter chacune des instructions suivantes :

```
a="étudiant-du-bts-services-informatiques-aux-organisations"
print(len(a))

print("*****")
#les caractères sont numérotés de gauche à droite 0,1,2,...,n :
print(a[0])
print(a[1])
print(a[55])
print(a[54])

print("*****")
#extractions :
print(a[:8])
print(a[::2])
print(a[16:])
print(a[12:15])#dans une tranche [p:q] le p-ième caractère est inclus mais pas le q-ième

print("*****")
#parcours de droite à gauche :
print(a[::-1])
print(a[::-2])

print("*****")
#parcours d'une chaîne :
for car in a :
    print(car + ' ', end = '')
```

## Exercice 3 : trouver l'indice d'un caractère

Ecrire une fonction « trouver » qui retrouve l'indice (ou les indices, c'est-à-dire le(s) numéro(s)) correspondant à un caractère présent dans une chaîne, ces données étant demandées à l'utilisateur. Cette fonction aura donc deux arguments : le nom de la chaîne à traiter et le caractère à trouver, et devra fournir en retour l'indice (ou les indices) du caractère de ce type dans la chaîne.

Ainsi, par exemple, l'instruction

```
print(trouver("Roméo et Juliette","é"))
```

devra afficher : 4.

#### Exercice 4 : compter le nombre de mots

Ecrire une fonction « comptermots » qui compte le nombre de mots dans une chaîne fournie par l'utilisateur (on notera qu'il suffit de compter les espaces).

Ainsi, par exemple, l'instruction

```
print(comptermots("les petits ruisseaux font les grandes rivières"))
```

devra afficher : 7.

#### Exercice 5 : compter un caractère donné

Ecrire une fonction « compter » qui compte le nombre d'occurrences d'un caractère dans une chaîne, ces données étant demandées à l'utilisateur. Cette fonction aura donc deux arguments : le nom de la chaîne à traiter et le caractère à compter.

Ainsi, par exemple, l'instruction

```
print(compter("ananas au jus","a"))
```

devra afficher : 4.

#### Exercice 6 : palindrome

Une chaîne est un palindrome si elle se lit de la même façon de gauche à droite et de droite à gauche, comme, par exemple, le mot « radar » ou le nombre 1 256 521.

Ecrire une fonction « estPalindrome » qui répond Vrai si la chaîne entrée par l'utilisateur est un palindrome.

#### Exercice 7 : nombres de Lychrel

On prend un entier naturel (à plusieurs chiffres) quelconque, par exemple  $n_1 = 524$ , et on l'écrit « à l'envers », comme ceci : 425. Si on ajoute ces deux nombres on obtient  $n_2 = 949$  qui est un nombre palindrome. On a observé qu'en itérant ce procédé autant de fois que nécessaire, tous les entiers fournissent un nombre palindrome, sauf si on prend un « nombre de Lychrel », qui est une exception à ce constat, comme par exemple  $n = 196$ .

Ecrire une fonction « testLychrel », qui prend comme arguments un entier  $n$  et un nombre maximal d'itérations  $maxIt$  fournis par l'utilisateur, et fournit le palindrome obtenu après le nombre d'itérations indiqué sinon répond qu'il y a échec et que le nombre  $n$  est susceptible d'être un nombre de Lychrel.