

Exercice 1

Écrivez les procédures `rotationGauche(tab)` et `rotationDroite(tab)` qui prennent un tableau en paramètre. Dans le cas de `rotationGauche` tous les éléments du tableau sont déplacés d'une position vers la gauche, sauf l'élément en position 0 qui est déplacé en dernière position. `rotationDroite` déplace les éléments vers la droite.

```
t = [1, 2, 3, 4, 5, 6]
```

```
rotationGauche(t); print(t) # affiche [2, 3, 4, 5, 6, 1]
rotationGauche(t); print(t) # affiche [3, 4, 5, 6, 1, 2]
rotationGauche(t); print(t) # affiche [4, 5, 6, 1, 2, 3]
rotationDroite(t); print(t) # affiche [3, 4, 5, 6, 1, 2]
```

Exercice 2

Écrivez la fonction `shuffle(tab)` qui prend un tableau en paramètre et retourne un nouveau tableau qui contient tous les éléments du tableau `tab` à des indexes pairs suivis par tous les éléments du tableau `tab` à des indexes impairs. Qu'est-ce qui est affiché par le programme suivant?

```
t = [1, 2, 3, 4, 5, 6, 7, 8]
```

```
print(t)
print(shuffle(t))
print(shuffle(shuffle(t)))
print(shuffle(shuffle(shuffle(t))))
```

Exercice 3

Écrivez une fonction `majuscule(texte)` qui prend en paramètre un texte et qui retourne un texte avec les mêmes caractères, sauf que le premier caractère, si c'est une lettre minuscule, sera remplacée par la lettre majuscule correspondante. Le comportement doit être le suivant :

```
print(majuscule('')) # affiche ligne vide
print(majuscule('x')) # affiche X
print(majuscule('42')) # affiche 42
print(majuscule('allo')) # affiche Allo
print(majuscule('Bonjour')) # affiche Bonjour
```

Exercice 4

Écrivez une fonction `decoderHexa(texte)` qui prend en paramètre un texte et qui retourne la valeur de ce texte interprété dans la notation hexadécimale de Python (qui commence par `0x`).

Si le texte ne respecte pas la notation hexadécimale (incluant la présence de blancs) la fonction doit retourner -1. Écrivez aussi des tests unitaires.

Exercice 5

Écrivez un programme qui décompose un texte en mots et affiche combien de fois apparaît chaque mot dans le texte. Par exemple,

```
analyser('il a un chat noir et un chat blanc')
```

doit imprimer :

```
il 1  
a 1  
un 2  
chat 2  
noir 1  
et 1  
blanc 1
```