

Projet Python – César n°1 – **correction**

Nom et prénom :

---

*Le chiffrement de César*

---

Le chiffrement de César est une technique cryptographique permettant de transformer un texte et de le rendre incompréhensible pour quiconque ne connaît pas la clef secrète. Le principe est simple : on choisit un nombre entre 1 et 26 et on décale les lettres de ce nombre en suivant l'ordre alphabétique. Le tableau ci-dessous montre l'évolution des lettres avec un décalage de 3.

Alphabet en clair	a b c d e f g h i j k l m n o p q r s t u v w x y z
Alphabet chiffré	d e f g h i j k l m n o p q r s t u v w x y z a b c

Ce qui donne :

message en clair	L'informatique c'est le futur.
messag chiffré	O'lqirupdwltxh f'hvw oh ixwxu.

Pour déchiffrer un message, il suffit de faire le décalage inverse.

**EXERCICE 1 :** Avec un décalage de 3, chiffrer le premier message et déchiffrer le deuxième.

Message en clair	Bonjour	Message en clair	Au revoir
Message chiffré	Erqmrxu	Message chiffré	Dx uhyrlu

---

*Fonctions utiles*

---

Pour rappel, pour parcourir un texte et en construire un nouveau, symbole par symbole, on peut utiliser une boucle, comme dans la fonction copie(texte).

```
def copie(texte):  
    resultat = ""  
    for symbole in texte:  
        resultat = resultat + symbole  
    return resultat
```

Vous pouvez utiliser la fonction upper() qui permet de transformer un texte en majuscules, de la manière ci-contre. Même si cette fonction peut gérer les lettres accentuées, nous n'en mettrons pas dans les textes. Nous pourrions par contre laisser les espaces et ponctuations.

```
>>> "Il était une fois.".upper()  
'IL ÉTAIT UNE FOIS.'  
>>> symbole = "a"  
>>> symbole.upper()  
'A'
```

Pour tout le sujet, on utilisera la variable globale ci-dessous :

```
ALPHABET = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
```

Pour obtenir la position d'une lettre ou obtenir la lettre à partir de sa position, il faut faire :

```
>>> ALPHABET.index("A")  
0  
>>> ALPHABET.index("M")  
12
```

```
>>> ALPHABET[0]  
'A'  
>>> ALPHABET[12]  
'M'
```

Pour décaler une lettre dans l'alphabet, on ajoute le décalage à sa position et on regarde la lettre correspondant. Puisque le nombre obtenu peut être supérieur à 26 ou inférieur à 0, il faut faire un modulo 26 :

```
>>> ALPHABET[(25+10)%26] # 35%26 -> 9  
'J'
```

```
>>> ALPHABET[(3-5)%26] # (-2)%26 -> 24  
'Y'
```

---

## Programmation

---

La fonction `decale(symbole, nb)` renvoie le symbole décalé de `nb` dans l'alphabet, si c'est une lettre majuscule. Sinon, on le renvoie inchangé. C'est le cas pour les espaces ou la ponctuation.

<pre>&gt;&gt;&gt; decale("G", 3) 'J'</pre>	<pre>&gt;&gt;&gt; decale("G", -3) 'D'</pre>	<pre>&gt;&gt;&gt; decale("!", 3) '!'</pre>	<pre>&gt;&gt;&gt; decale("A", -3) 'X'</pre>
--	---	--	---

**EXERCICE 2 :** Compléter les phrases ci-dessous :

- |  |  |
|--|--|
| 1) <code>decale("A", 7)</code> renvoie 'H'.  | 4) <code>decale("M", -2)</code> renvoie 'K'. |
| 2) <code>decale("C", -4)</code> renvoie 'Y'. | 5) <code>decale("R", -6)</code> renvoie 'L'. |
| 3) <code>decale("M", 4)</code> renvoie 'Q'.  | 6) <code>decale(";", 14)</code> renvoie ';'. |

**EXERCICE 3 :** Compléter le code de la fonction ci-dessous :

```
def decale(symbole, nb):
    if symbole in ALPHABET:
        i = ALPHABET.index(symbole)
        j = (i + nb) % 26
        return ALPHABET[j]
    else:
        return symbole
```

**EXERCICE 4 :** Compléter la fonction `chiffre(texte, nb)` qui renvoie un nouveau texte où toutes les lettres ont été décalées par `nb`. Les autres symboles ne sont pas modifiés. Les symboles sont passés en majuscule avant l'appel à `decale`.

```
>>> chiffre("avion", 5)
'FANTS'
```

```
>>> chiffre("Le chien.", 10)
'VO MRSOX.'
```

```
def chiffre(texte, nb):
    resultat = ""
    for symbole in texte:
        resultat = resultat + decale(symbole.upper(), nb)
    return resultat
```

**EXERCICE 5 :** Mettre la bonne expression pour que la fonction `dechiffre(texte, nb)` renvoie un nouveau texte correspondant au message déchiffré, alors que ce message avait été chiffré avec un décalage de `nb`.

```
>>> dechiffre('FANTS', 5)
'AVION'
```

```
>>> dechiffre('VO MRSOX.', 10)
'LE CHIEN.'
```

```
def dechiffre(texte, nb):
    return chiffre(texte, -nb)
```

**EXERCICE 6 :** Décoder le message suivant et **répondre à la question**. À vous de trouver le décalage.

'HLVC AFLI VJK DFIK ALCVJ TVJRI ?' Il est mort le 15 mars 44 av. J.-C.