# Challenge 2024-06-22 : Cryptage symmétrique

Table des matières

[Challenge 2024-05-31 : Classement par moyennes 1](#_Toc168519328)

[1) Enoncé 3](#_Toc168519329)

[2) Logigramme des modules 4](#_Toc168519330)

[a. Logigramme simple du code 4](#_Toc168519331)

[3) Variables du code 5](#_Toc168519332)

[4) Code 6](#_Toc168519333)

[b. constantes.py 6](#_Toc168519334)

[c. main.py 7](#_Toc168519335)

[1) Résulta **Erreur ! Signet non défini.**](#_Toc168519336)

## Enoncé

👉 Le but de ce challenge est de développer un codeur/décodeur suivant la logique de [cryptographie symétrique](<https://fr.wikipedia.org/wiki/Cryptographie_sym%C3%A9trique>) qui utilise une clef numérique très simple pour chiffrer et déchiffrer.

🔹 \*\*Étapes\*\*

1. Envoyer la phrase à chiffrer : `Salut, je suis ici pour apprendre Python` et la clef de codage : 6

2. On remplace d'abord tous les espace par des "\_" -> `Salut,\_je\_suis\_ici\_pour\_apprendre\_Python`

3. Comme la clef est 6, on découpe la phrase en ligne de 6 caractères, les cases manquants sont comblés par des " \* " :

Salut,

\_je\_su

is\_ici

\_pour\_

appren

dre\_Py

thon\*\*

4. On transpose en lisant le texte par colonne (de haut en bas et de gauche à droite) donc suivant cet ordre :

1 8 15 22

2 9 16 23

3 10 17 24

4 11 18 25

5 12 19 26

6 13 20 27

7 14 21 28

La phrase chiffrée sera donc la suivante :

`S\_i\_adtajspprhle\_opeou\_iur\_ntscreP\*,ui\_ny\*`

\*\*Pour le décodage, il suffira de suivre les étapes dans le sens inverse...\*\*

🔹 \*\*Conditions\*\*

- L'affichage se fait via la console

- La phrase a chiffrer est a envoyer en argument, accompagnée de sa clef au format int, le résultat doit retourner la phrase codée

- Pour le décodeur, la phrase chiffrée est envoyée accompagnée de la clef, le résultat doit retourner la phrase en clair

- La clef doit être la même pour un même chiffrage et déchiffrage

- Vous êtes libre de coder en fonctionnel ou bien constituer vos classes en POO

🔹 \*\*Exemples\*\*

\*Chiffrage\*

- `Lorem ipsum dolor sit amet, consectetur adipiscing elit.` clef : 12 -> `Ldetnootugrl,r\_eo\_\_emrcal\_\_odiisnitpisp.stei\*u\_cs\*matc\*\_mei\*`

- `La première machine programmable a été réalisé en 1801.` clef : 3 -> `Lpmrmherrmlatrlén8.arieai\_oaae\_ééi\_\_0\*\_eè\_cnpgmb\_é\_ase11\*`

\*Déchiffrage\*

- `Lag\_eeelelm\*\_l\_aa\*ced\_i\*hnesn\*` clef : 5 -> `Le challenge de la semaine`

- `dctigosrn\*` clef : 2 -> `docstring`

## Logigramme des modules

### Logigramme simple du code

## Variables du code

|  |  |  |  |
| --- | --- | --- | --- |
| Modules | Nom de la variable | Type | Commentaires |
|  |  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |  |

## Code

### constantes.py

### main.py

## Résultat