

PROJET SUR LES CRYPTAGES : MODE D'EMPLOI

1. STRUCTURE DES FICHIERS

Pour utilisation :

cryptage /

```
...                // reste des fichiers accompagnant l'exécutable  
output/            // contiendra les fichiers.txt des métadonnées  
cryptage.exe       // fichier exécutable  
icone.ico          // icône du logiciel  
...
```

Pour développement :

src/

modules/

algorithmes/

__init__.py

cesar.py

polybe.py

rot13.py

vigenere.py

Ces fichiers d'extension ".py"
contiennent les fonctions de
codage et décodage selon la
méthode du nom du fichier. Ils
sont importés et utilisés dans le
fichier principal pour réaliser les
cryptages et les montrer sur
l'interface, mais ils peuvent être
utilisés séparément.

interface/

__init__.py

interface.py

DE-LA-CUEVA_Daniel_code.py // fichier principal, en l'exécutant l'interface apparaît

2. PRÉREQUIS

Langage :

- Python 3, doit être installé si on veut exécuter le fichier « .py ».
Version utilisée : Python 3.10

Bibliothèques tierces utilisées :

- Tkinter (pip install tk) : à installer si on exécute le code depuis le fichier « .py », non nécessaire si on exécute le fichier « .exe ».
- PyInstaller : utilisée pour produire le fichier « .exe », elle n'est pas nécessaire au fonctionnement du code.

3. FONCTIONS DE CODAGE

Les fonctions utilisées au sein du projet pour réaliser l'action de crypter et décrypter en soi sont contenues dans des fichiers indiquant le nom de l'algorithme utilisé. Chaque fichier contient une fonction « _c » (codage) et une fonction « _d » (décodage).

Exemple : Le fichier « cesar.py » contient les fonctions « cesar_c » et cesar_d ».

Les différentes fonctions prennent différents arguments, mais retournent toutes un dictionnaire à forme similaire :

FONCTION	ARGUMENTS	RETOURNE
cesar_c	<ul style="list-style-type: none">- message_clair (str)- decalage (int), 3 par défaut	<p>Un dictionnaire contenant :</p> <ul style="list-style-type: none">- méthode- alphabet utilisé- chiffres utilisés- clé (si elle existe)- message clair- message codé <p>Types de données possibles :</p> <ul style="list-style-type: none">- str- int- NoneType
cesar_d	<ul style="list-style-type: none">- message_code (str)- decalage (int), 3 par défaut	
rot13_c	<ul style="list-style-type: none">- message_clair (str)	
rot13_d	<ul style="list-style-type: none">- message_code (str)	
vigenere_c	<ul style="list-style-type: none">- message_clair (str)- cle_orig (str), clé de chiffrement- alphabet (str), alphabet latin par défaut	
vigenere_d	<ul style="list-style-type: none">- message_code (str)- cle_orig (str), clé de chiffrement- alphabet (str), alphabet latin par défaut	

polybe_c	<ul style="list-style-type: none">- message_clair (str)- alphabet (str), alphabet latin par défaut	
polybe_d	<ul style="list-style-type: none">- message_code (str)- alphabet (str), alphabet latin par défaut	

Si on exécute les fichiers contenant les fonctions, un message nous indiquera le fichier que nous avons choisi mais il n'y aura possibilité de faire aucune opération. Il faut donc importer ces modules pour les utiliser.

Note : Dû aux restrictions de la programmation à notre niveau, et pour éviter de compliquer les fonctions excessivement, certaines règles sont appliquées :

- Le message traité sera toujours renvoyé en lettres majuscules, quelle que soit la case à l'entrée
- L'alphabet par défaut est l'alphabet latin et les chiffres de 0 à 9, il ne peut être changé sur l'interface mais l'utilisateur plus avancé peut utiliser les fonctions indépendamment et utiliser un autre alphabet.
- Les autres caractères (espaces, virgules, lettres accentuées) ne sont pas codés, ils conservent leur position dans le message.
- Les chiffres ne sont pas codés avec Vigenère, ils restent tels qu'ils sont. La clé de codage ne peut contenir des nombres sous cette méthode également.
- Les chiffres ne peuvent pas être codés avec Polybe, car ils seraient confondus avec le message chiffré.
- L'alphabet doit comporter 25 caractères (5x5) pour le chiffre de Polybe.

4. INTERFACE

Voici l'interface graphique du logiciel :

The screenshot shows a window titled 'Encodeur - Décodeur'. At the top, it says 'Choisissez la méthode, le mode et la clé de codage'. Below this are two radio buttons: 'Encoder' (selected) and 'Décodeur'. A box labeled 'CHOIX DU MODE' points to these buttons. Below the radio buttons is a button labeled 'CODE DE VIGENÈRE' with a dropdown arrow. A box labeled 'CHOIX DE LA MÉTHODE' points to this button. Below that is a text input field labeled 'Entrez le message:' containing the text 'J'AI REUSSI'. A box labeled 'INTRODUCTION DU MESSAGE' points to this field. Below the message field is another text input field labeled 'Entrez la clé de codage:' containing the text 'PROJET'. A box labeled 'INTRODUCTION DE LA CLÉ' points to this field. Below the key field is a text output area labeled 'Le résultat est:' containing the text 'Y'RW AINHJW'. A box labeled 'MÉSSAGE TRAITÉ' points to this area. At the bottom right are two buttons: 'Coder' and 'Sauvegarder'. A box labeled 'BOUTON DE CODAGE' points to the 'Coder' button. A box labeled 'BOUTON DE SAUVEGARDE:' points to the 'Sauvegarder' button.

BOUTON DE CODAGE

BOUTON DE SAUVEGARDE:

Un fichier avec la date et les données de l'opération sera créé dans le repertoire /output

Plus de détails : https://github.com/DanieldelaCueva/NSI_FILES_2122