



Cryptographie - l'art de cacher

Mini Projet : Message secret

Aide au décryptage complet du message

Ce sujet apporte une aide au décryptage du message codé par le code de Che Guevara. La méthode utilisée sera basée sur l'utilisation de fonctions. L'organisation des fonctions utilisées est décrite en page 3.

La bibliothèque de fonctions incomplète est disponible dans le fichier `decryptageCheGuevara.py`. Elle est donc à compléter en suivant la démarche proposée suivante.

DÉCRYPTAGE SELON LA TABLE ALPHABÉTIQUE

La fonction `decoupeTexteCrypte(message, tableAlphabet)` présente dans la bibliothèque `decryptageCheGuevara.py` est incomplète. La description suivante décrit son fonctionnement :

Nom de la fonction : **`decoupeTexteCrypte(message, tableAlphabet)`**

But : Découper les caractères selon la table alphabétique.

Arguments :

- ➔ *message* : Texte crypté moins la clé (type string)
- ➔ *Table alphabet* : Liste des codes alphabétiques (liste de string)

Retour :

- ➔ Listes des valeurs codées découpées selon l'alphabet (liste de string)

Exemple :

```
>>>decoupeTexteCrypte('830787249',['6','38','32','4','8','30','36',  
'34','39','31','78','72','70','76','9','79','71','58','2','0','52','50','56','54','1','59'])  
['8','30','78','72','4','9']
```

- x **Compléter** la fonction `decoupetexteCrypte` afin qu'elle respecte le fonctionnement décrit ci-dessus. Les zones à compléter sont repérés dans la définition de la fonction sous forme de commentaires.
- x **Décrire** un test possible pour valider son fonctionnement. **Valider** le fonctionnement en appliquant ce test.



Cryptographie - l'art de cacher

Mini Projet : Message secret

DÉCRYPTAGE DE LA CLÉ

- x **Compléter** la fonction de décryptage de la clé `decryptCle(texteCrypte,cle)` en suivant les remarques ci-dessous. Cette fonction doit répondre aux caractéristiques suivantes :

Nom de la fonction : **`decryptCle(texteCrypte,cle)`**

Decrypte le message avec la clé de cryptage.

Arguments d'entrée :

→ *texteCrypte* : texte crypté (type string) avec les espaces

→ *cle* : Clé de cryptage (type string)

Valeur de sortie :

→ Texte décrypté (type string)

Exemple :

```
>>decryptCle('16843','5555')  
'61398'
```

Remarques : Pour cette fonction il est nécessaire d'effectuer des opérations mathématiques (soustractions) pour retrancher la clé au message codé. Ces opérations ne pouvant pas être réalisées sur des chaînes de caractères, il est donc nécessaire d'effectuer une modification du type pour convertir les caractères ASCII représentant un chiffre en un chiffre entier. Cette opération est effectuée avec l'instruction `int()`.

Exemple

`int('2')` renvoie la valeur 2.

Donc l'opération `'2' + '2'` renvoie la chaîne de caractères `'22'` (concaténation de chaînes)

Alors que l'opération `int('2') + int('2')` renvoie donc la valeur 4 (addition)

L'opération inverse est réalisée avec l'instruction `str()`

Exemple :

`str(22)` renvoie la chaîne de caractères `'22'`

DÉCRYPTAGE COMPLET DU MESSAGE

Pour utiliser la bibliothèque que vous venez de compléter, il est nécessaire de placer ce fichier dans le même répertoire que votre fichier script et de commencer votre script par l'instruction d'importation suivante :

```
from decryptageCheGuevara import *
```

- x **Ecrire** et tester un programme python mettant en œuvre les fonctions `decryptCle`, `decoupeTexteCrypte` et `decrypteAlphabet`, afin de décrypter le message codé.

Il est conseillé de tester chaque fonction indépendamment les unes des autres sur la console avec de bien comprendre leur fonctionnement.



Cryptographie - l'art de cacher

Mini Projet : Message secret

PRINCIPE DE DÉCODAGE DU CODE DE CHE GUEVARA

