MPSI 2017-2018

Informatique

TP 4 : cryptographie monoalphabétique

La cryptographie monoalphabétique consiste à transformer chaque lettre d'un texte en une autre lettre à l'aide d'une "clé" pour que ce texte devienne incompréhensible pour les personnes ne connaissant pas la clé. Plus précisément si A désigne l'ensemble des lettres de l'alphabet :

$$A = \{a, b, c, \ldots, z\}$$

la clé est une bijection ϕ de A dans A et le texte clair "$a_1a_2a_3\ldots a_n$" est codé en "$\phi(a_1)\phi(a_2)\phi(a_3)\ldots\phi(a_n)$".

Partie I. Implémentation de ϕ.

On va coder ϕ par un tableau associatif ou dictionnaire. Le dictionnaire fonctionne comme une liste si ce n'est qu'on accède aux éléments à l'aide d'un identifiant plutôt que par leur position. Les identifiants peuvent être des chaines de caractères, des couples, des entiers... Exemple :

```
>>> MaListe = [12, 20, 13]
>>> MaListe[0]
12

>>> MonDico = {'Pierre' : 12, 'Luc' : 20, 'Atila' : 13}
>>> MonDico['Pierre']
12
```

La clé sera donc un dictionnaire. Voici des exemples :

```
Clé1 = {'a' : 'd', 'b' : 'e', 'c' : 'f', ..., 'w' : 'z', 'x' : 'a', 'y' : 'b', 'z' : 'c'}
Clé2 = {'a' : 'z', 'b' : 'w', 'c' : 'k', ...}
```

Ainsi dans l'exemple 1, l'image de 'a' est 'd', l'image de 'b' est 'e', l'image de 'c' est 'f'... Ce chiffrement obtenu en décalant les lettres de 3 rangs est appelé chiffrement de César. Dans l'exemple 2, l'image de 'a' est 'z', l'image de 'b' est 'w', l'image de 'c' est 'k'...

1. Écrire une fonction qui construit une clé qui décale toutes les lettres de n rangs. La syntaxe devra être :

   CléCesar(n)

2. Écrire une fonction qui construit une clé aléatoire. La syntaxe devra être :

   CléHasard()

3. Écrire une fonction qui à partir d'une clé représentant une bijection ϕ donne la clé représentant la bijection ϕ^{-1}. La syntaxe devra être :

   CléInverse(Clé)

Partie II. Codage/Décodage

1. Écrire une fonction qui code une lettre suivant une clé. La syntaxe devra être :

   CodeLettre(Lettre, Clé)

   Elle reverra la lettre codée. Cette fonction devra gérer entre autre :
   - les lettres majuscules même si la clé ne contient que des minuscules,
   - les accents et autres signes diacritiques classiques : à ç é è ê î ô ù ; ces lettres particulières devront être codées comme la lettre non modifiée correspondante,
   - les caractères non alphabétiques ne devront pas être modifiés.

2. Écrire une fonction qui décode une lettre suivant une clé. La syntaxe devra être :

   DecodeLettre(Lettre, Clé)

3. Écrire une fonction qui code une chaine de caractère suivant une clé. La syntaxe devra être :

   CodeChaine(Lettre, Clé)

   Elle devra bien sûr appeler la fonction *CodeLettre*.

4. Écrire une fonction qui décode une chaine de caractère suivant une clé. La syntaxe devra être :

   DecodeChaine(Lettre, Clé)

   Elle devra bien sûr appeler la fonction *DecodeLettre*.

Partie III. Manipulation de fichiers

Les textes en clair et codé seront dans des fichiers externes, il faudra donc pouvoir lire et écrire dans ces fichiers depuis votre programme. Cela se fait en 3 étapes :

- On ouvre le fichier avec la commande *objFich = open(NomFichier, Type)* où
  - *NomFichier* est le nom de votre fichier avec le chemin d'accès. Pour éviter les problèmes de chemin d'accès, vous mettrez vos fichiers à la racine de votre clé USB. Ainsi pour accéder à un fichier de nom *NomFic.txt*, il faudra mettre *F :/NomFic.txt*.
  - *Type* est le type d'ouverture : 'r' (read) / 'w' (write) pour une ouverture en lecture seule / écriture seule.
  - *objFich* est la variable contenant l'objet renvoyé par la fonction.
- On lit ou on écrit dans le fichier avec les instructions :

  objFich.write(TexteAEcrire),
  TexteLu = objFich.read().

- On ferme le fichier avec l'instruction *objFich.close()*.

1. Écrire une fonction qui lit dans un fichier un texte en clair que vous aurez écrit (100 mots minimum), le crypte et l'écrit dans un fichier qui aura le même nom que le fichier d'origine avec un suffixe '_Code' (Exemple le fichier 'MonFichier.txt' deviendra 'MonFichier_Code.txt'. Attention de ne pas mettre dans ce fichier des caractères que vous n'avez pas gérés dans la question I.1.. La syntaxe devra être :

   CodeFichier(NomFichier, Clé)

2. Écrire une fonction qui lit dans un fichier un texte codé, le décode avec la clé l'écrit dans un fichier qui aura le même nom que le fichier d'origine avec un suffixe '_Decode' (Exemple le fichier 'MonFichier_Code.txt' deviendra 'MonFichier_Code_Decode.txt'. La syntaxe devra être :

DecodeFichier(NomFichier, Clé)

Partie IV. Cryptanalyse. Le but de cette partie est de développer des outils permettant de décoder un texte codé de la manière précédente sans la clé.

1. Écrire une fonction qui calcule la fréquence d'apparition de chaque lettre dans un texte codé contenu dans un fichier. La syntaxe devra être :

   Frequence(NomFichier)

   Cette fonction devra renvoyer un tableau associatif du type : {'a' : 0.02, 'b' : 0.0143, ...}.

2. Télécharger le fichier se trouvant à l'adresse 'www.ericreynaud.fr/Tel/DicoFr.txt' et le copier sur sa clé USB. Ce fichier contient les mots de la langue française. Ecrire une (ou plusieurs) procédure(s) permettant à partir d'une chaine de caractères contenant des lettres et le symbole *, de connaître s'il existe un mot de la langue française qui peut être obtenu en remplaçant les * par des lettres. Par exemple 'Bon**ur' devra renvoyer la liste ['Bonjour'].

3. Pour les plus rapides, écrire une fonction permettant de décoder un texte automatiquement.