

EXERCICE 2 (4 points)

Cet exercice porte sur les structures de données (dictionnaires).

La cryptographie est un ensemble de techniques permettant de chiffrer un message.

Une technique de cryptographie consiste à mélanger les lettres d'un alphabet et à réécrire le message avec ces permutations. En Python, on peut créer un dictionnaire dans lequel les clés sont les lettres de l'alphabet et les valeurs sont celles de l'alphabet mélangé.

Par exemple, si l'alphabet contient les 4 lettres A, B, C et D, et si le dictionnaire de l'alphabet mélangé est `alpha = {"A": "B", "B": "D", "C": "A", "D": "C"}`, la chaîne de caractères "BAC" sera chiffrée "DBA".

Un tel dictionnaire sera appelé **dictionnaire de chiffrement**.

1. On souhaite chiffrer un message écrit avec l'alphabet A, B, C, D, E, F, G à l'aide du dictionnaire `alpha = {"A": "B", "B": "D", "C": "A", "D": "C", "E": "F", "F": "G", "G": "E"}`
 - a. Quelle est la valeur associée à la clé "D" ? En Python, comment l'obtenir ?
 - b. Chiffrer la chaîne de caractères "BAGAGE" avec le dictionnaire `alpha`.
2. On considère qu'un mot est une chaîne de caractères (un objet de type `str`) écrite uniquement avec les 26 lettres de l'alphabet en majuscule. Par exemple, "ARBRE" est un mot et "L' ARBRE !" n'est pas un mot à cause des caractères : "'", " "(espace) et "!".
Écrire une fonction `chiffrer(mot, alpha)` qui prend en paramètres `mot` un mot et `alpha` un dictionnaire de chiffrement, et qui renvoie une chaîne de caractères chiffrée avec le dictionnaire de chiffrement `alpha`.
3. On souhaite déchiffrer un mot chiffré avec cette méthode.
 - a. Si un mot est chiffré avec le dictionnaire de chiffrement `alpha = {"A": "B", "B": "D", "C": "A", "D": "C", "E": "F", "F": "G", "G": "E"}`, donner un dictionnaire permettant de le déchiffrer.
 - b. Ecrire une fonction en Python appelée `dico_dechiffrement(dico)` qui prend en paramètre `dico` un dictionnaire de chiffrement et qui renvoie un dictionnaire permettant le déchiffrement. On pourra s'inspirer du code incomplet ci-dessous ou proposer une autre solution :

```
def dico_dechiffrement(dico):
    nouveau = {}
    for lettre in dico :
        code = dico[.....]
        nouveau[.....] = .....
    return nouveau
```

- c. Ecrire une fonction `dechiffre(mot, dico)` qui reçoit un mot chiffré et un dictionnaire de chiffrement et renvoie le mot décodé. On utilisera les fonctions écrites dans les questions précédentes.

4. On souhaite à présent créer un dictionnaire de chiffrement. Ecrire une fonction `dico_chiffrement(alphabet)` qui prend en paramètre `alphabet` un tableau de lettres et qui renvoie un dictionnaire de chiffrement dont les clés sont les lettres du tableau `alphabet` et les valeurs sont les lettres du tableau `alphabet` mélangées.

On pourra utiliser la fonction `shuffle` du module `random` qui mélange en place un tableau. Par exemple, on a :

```
>>> tab = ["A", "B", "C", "D"]
>>> shuffle(tab)
>>> tab
["B", "A", "D", "C"]
```