

Concours blanc Juin 2019 – Partie écrite. Durée 1 h.

Lorsqu'on écrit un code Python, on fait attention à ce que les indentations soient visibles sur la copie.

I Chiffrement de Vigenère

La plupart des questions sont indépendantes.

On considère un texte initial sans aucune majuscule, ni accent ni signes diacritiques (accents, tildes, etc.). Les mots sont exclusivement séparés par des espaces (pas d'apostrophes, de tirets, de guillemets, etc.) et les phrases se terminent par un point. Il n'y a aucun autre signe de ponctuation (ni virgule, ni point-virgule, par exemple) et le texte est écrit en un seul bloc (pas de saut de ligne, etc.).

On souhaite coder ce texte grâce au chiffrement de Vigenère. Pour cela :

- A** On prépare le texte à chiffrer en éliminant toutes les espaces et tous les points.
- B** On associe à chaque lettre un entier, selon le principe suivant : la lettre a est associée à l'entier 0, la lettre b est associée à l'entier 1, etc., la lettre z est associée à l'entier 25.
- C** On chiffre le contenu, c'est-à-dire qu'on le code grâce à une clé de chiffrement. Le principe est le suivant :
 - On choisit un mot-clé qui sera la clé de chiffrement.
 - On détermine les entiers qui correspondent aux lettres de ce mot-clé, selon le même principe que pour le texte à chiffrer.
 - On décale le premier entier de la liste qui correspond au texte à chiffrer d'une valeur qui est égale à l'entier qui correspond à la première lettre du mot-clé.
 - On décale le deuxième entier de la liste qui correspond au texte à chiffrer d'une valeur qui est égale à l'entier qui correspond à la deuxième lettre du mot-clé.
 - Etc.
 - Quand on a utilisé tous les entiers qui correspondent au mot-clé, on recommence au début du mot-clé.
- D** On forme le texte chiffré par l'opération inverse de l'étape **B** : l'entier 0 est associé à la lettre a. Etc. Pour tenir compte du décalage des entiers, on associe évidemment 26 à la lettre a, 27 à la lettre b, etc.

Exemple : texte à chiffrer : `lycee dorian.` et mot-clé : `ptsi`

texte initial	l	y	c	e	e		d	o	r	i	a	n	.
texte à chiffrer	l	y	c	e	e	d	o	r	i	a	n		
entiers associés au texte à chiffrer	11	24	2	4	4	3	14	17	8	0	13		
mot-clé	p	t	s	i	p	t	s	i	p	t	s		
chiffres associés au mot-clé	15	19	18	8	15	19	18	8	15	19	18		
entiers décalés du texte à chiffrer	26	43	20	12	19	22	32	25	23	19	31		
texte chiffré	a	r	u	m	t	w	g	z	x	t	f		

0. Donner le texte chiffré correspondant au texte `jaimebeaucoup` chiffré à l'aide du mot-clé `linfo`

On considère que l'environnement python connaît déjà une variable de type `list`, nommée `alphabet` et contenant la liste de toutes les lettres de l'alphabet, sous forme de caractères de type `str` :

`alphabet=['a','b','c','d','e','f','g','h','i',[...], 'r','s','t','u','v','w','x','y','z']`

1. Proposer le code python d'une fonction `nettoyage` prenant en entrée un texte sous la forme d'une variable de type `str` et renvoyant une variable de type `list` contenant une liste des lettres, et uniquement les lettres, qui composent le texte.
2. Donner le code d'une fonction `lettres_vers_entiers` prenant en entrée une liste de lettres et retournant une liste des entiers associés à ces lettres.
3. Écrire une fonction `chiffrement_vigenere(L,C)` qui prend en entrée une liste d'entiers (L) associés à un texte à chiffrer et une liste d'entiers (C) associés au mot-clé et qui retourne une liste des entiers décalés du texte à chiffrer.
4. Donner le code d'une fonction `entiers_vers_lettres` prenant en entrée une liste d'entiers et retournant la liste des lettres associées à ces entiers.
5. Proposer le script écrit en python qui, en utilisant chacune des fonctions précédentes (qu'on supposera connues), permet d'afficher à l'écran le texte chiffré correspondant à un texte contenu initialement dans la variable `texte_initial` de type `str`.

II Bases de données – Questions de cours

On dispose dans une base de données d’une table qui contient des informations datées de 2010 sur les communes de France nommée **ville_france** et dont le schéma relationnel est le suivant :

(ville_id :int, ville_departement :int, ville_region :int, ville_nom_reel :char, ville_code_postal :int, ville_population_2010 :int)

1. Donner le code de la requête SQL permettant d’afficher la totalité du contenu de la table.
2. Donner le code de la requête SQL permettant d’afficher le nombre de lignes de la table.
3. Donner le code de la requête SQL permettant d’afficher le nom et la population de toutes les villes dont le code postal est 40990.

Le résultat de la requête précédente est :

ville_nom_reel	ville_population_2010
Saint-Vincent-de-Paul	2997
Téthieu	656
Gourbera	356
Herm	1044
Mées	1736
Saint-Paul-lès-Dax	12409
Angoumé	288

4. Donner le code de la requête SQL permettant d’afficher la population totale cumulée des villes dont le code postal est 40990.

On dispose, dans la même base de données, de deux tables supplémentaires, contenant les noms des départements et des régions.

- La table **departement**, de schéma relationnel (departement_id :int, nom_departement :char)
 - La table **region**, de schéma relationnel (region_id :int, nom_region :char)
5. Donner le code de la requête SQL permettant d’afficher le nom de toutes les villes, avec leur population, le nom de leur département et le nom de leur région.