Ave Cesar (d'après X2008 - MP)

Durée : 1 heure 30

On cherche à crypter un texte t de longueur n composé de caractères en minuscules (soit 26 lettres différentes) représentés par des entiers compris entre 0 et 25 ($0 \leftrightarrow$ a, $1 \leftrightarrow$ b,..., $25 \leftrightarrow$ z). Nous ne tenons pas compte des éventuels espaces.
Ainsi, le texte `ecolepolytechnique` est représenté par le tableau

```
[4, 2, 14, 11, 4, 15, 14, 11, 24, 19, 4, 2, 7, 13, 8, 16, 20, 4]
```

comme le montre le schéma ci dessous : la première ligne représente le texte, la seconde les entiers correspondants, et la troisième les indices dans le tableau t.

e	c	o	l	e	p	o	l	y	t	e	c	h	n	i	q	u	e
4	2	14	11	4	15	14	11	24	19	4	2	7	13	8	16	20	4
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

Partie I. Codage de César

Ce codage est le plus rudimentaire que l'on puisse imaginer. Il a été utilisé par Jules César (et même auparavant) pour certaines de ses correspondances. Le principe est de décaler les lettres de l'alphabet vers la gauche de 1 ou plusieurs positions. Par exemple, en décalant les lettres de 1 position, le caractère a se transforme en z, le b en a, ... le z en y. Le texte `avecesar` devient donc `zudbdrzq`.

Question 1. Que donne le codage du texte `maitrecorbeau` en utilisant un décalage de 5 ?

Question 2. Écrire la fonction `codageCesar(t, d)` qui prend en arguments le tableau t et un entier d ; et qui renvoie un tableau de même taille que t contenant le texte t décalé de d positions.

Question 3. Écrire de même la fonction `decodageCesar(t, d)` prenant les mêmes arguments mais qui réalise le décalage dans l'autre sens.

Pour réaliser ce décodage, il faut connaître la valeur du décalage. Une manière de la déterminer automatiquement est d'essayer de deviner cette valeur. L'approche la plus couramment employée est de regarder la fréquence d'apparition de chaque lettre dans le texte crypté. En effet, la lettre la plus fréquente dans un texte suffisamment long en français est la lettre e.

Question 4. Écrire la fonction `frequences(tprime)` qui prend en argument un tableau t' représentant le texte crypté ; et qui renvoie un tableau de taille 26 dont la case d'indice i contient le nombre d'apparitions du nombre i dans t' ($0 \leqslant i < 26$).

Question 5. Écrire la fonction `decodageAuto(tprime)` qui prend en argument le tableau t' représentant le texte crypté ; et qui renvoie le texte d'origine (en calculant la clef pour que la lettre e soit la plus fréquente dans le texte décrypté).

Partie II. Codage de Vigenère

Au XVI^{e} siècle, Blaise de Vigenère a modernisé le codage de César très peu résistant de la manière suivante. Au lieu de décaler toutes les lettres du texte de la même manière, on utilise un texte clef qui donne une suite de décalages.
Prenons par exemple la clef `concours`. Pour crypter un texte, on code la première lettre en utilisant le décalage qui envoie le a sur le c (la première lettre de la clef). Pour la deuxième lettre, on prend le décalage qui envoie le a sur le o (la seconde lettre de la clef) et ainsi de suite. Pour la huitième lettre, on utilise le décalage a vers s, puis, pour la neuvième, on reprend la clef à partir de sa première lettre. Sur l'exemple `ecolepolytechnique` avec la clef `concours`, on obtient : (la première ligne donne le texte, la seconde le texte crypté et la troisième la lettre de la clef utilisée pour le décalage)

e	c	o	l	e	p	o	l	y	t	e	c	h	n	i	q	u	e
g	q	b	n	s	j	f	d	a	h	r	e	v	h	z	i	w	s
c	o	n	c	o	u	r	s	c	o	n	c	o	u	r	s	c	o

Question 6. Donner le codage du texte `becunfromage` en utilisant la clef de codage `jean`.

Question 7. Écrire la fonction `codageVigenere(t, c)` qui prend comme arguments un tableau t représentant le texte à crypter et un tableau d'entiers c donnant la clef servant au codage ; et qui renvoie un tableau contenant le texte crypté t'.

Maintenant, on suppose disposer d'un texte t' assez long crypté par la méthode de Vigenère, et on veut retrouver le texte t d'origine. Pour cela, on doit trouver la clef c ayant servi au codage. On procède en deux temps : 1) détermination de la longueur k de la clef c, 2) détermination des lettres composant c.

La première étape est la plus difficile. On remarque que deux lettres identiques dans t espacées de $\ell \times k$ caractères (où ℓ est un entier et k la taille de la clef) sont codées par la même lettre dans t'. Mais cette condition n'est pas suffisante pour déterminer la longueur k de la clef c puisque des répétitions peuvent apparaître dans t' sans qu'elles existent dans t. Par exemple, les lettres `t` et `n` sont toutes deux codées par la lettre `h` dans le texte crypté à partir de `ecolepolytechnique` avec `concours` comme clef. Pour éviter ce problème, on recherche les répétitions non pas d'une lettre mais de séquences de lettres dans t' puisque deux séquences de lettres répétées dans t, dont les premières lettres sont espacées par $\ell \times k$ caractères, sont aussi cryptées par deux mêmes séquences dans t'.

Dans la suite de l'énoncé, on ne considère que des séquences de taille 3 en supposant que toute répétition d'une séquence de 3 lettres dans t' provient exclusivement d'une séquence de 3 lettres répétée dans t. Ainsi, la distance séparant ces répétitions donne des multiples de k.
La valeur de k est obtenue en prenant le PGCD de tous ces multiples. Si le nombre de répétitions est suffisant, on a de bonnes chances d'obtenir la valeur de k. On suppose donc que cette assertion est vraie.

Question 8. Écrire la fonction `pgcd(a, b)` qui calcule le pgcd des deux entiers strictement positifs a et b par soustractions successives de ses arguments.

Question 9. Écrire la fonction `pgcdDesDistancesEntreRepetitions(tprime, i)` qui prend en argument le texte crypté t' de longueur n et un entier i ($0 \leqslant i < n-5$) qui est l'indice d'une lettre dans t' ; et qui renvoie le pgcd de toutes les distances entre les répétitions de la séquence de 3 lettres $\langle t[i], t[i+1], t[i+2]\rangle$ dans la suite du texte $\langle t[i+3], t[i+4], \dots t[n-1]\rangle$. Cette fonction renvoie 0 s'il n'y a pas de répétition.

Question 10. Écrire la fonction `longueurDeLaClef(tprime)` qui prend en argument le texte crypté t' ; et qui renvoie la longueur k de la clef de codage.

Question 11. Calculer le nombre maximal d'appels à la fonction `pgcd` réalisés par la fonction `longueurDeLaClef` en fonction de la longueur n du tableau t'.

Question 12. Une fois la longueur de la clef connue, donner une idée d'algorithme permettant de retrouver chacune des lettres de la clef.
En déduire une fonction `decodageVigenereAuto(tprime)` qui prend en argument le tableau t' représentant le texte crypté ; et qui renvoie le texte t d'origine.

♦ ♦
♦