CHIFFREMENT AFFINE

Commentaire : Utiliser Python pour coder/décoder un message à l'aide de chiffrements affines.
En prérequis, il est souhaitable d'avoir traité cette activité (ou équivalente) : *https://www.maths-et-tiques.fr/telech/codage.pdf*

A	0
B	1
C	2
D	3
E	4
F	5
G	6
H	7
I	8
J	9
K	10
L	11
M	12
N	13
O	14
P	15
Q	16
R	17
S	18
T	19
U	20
V	21
W	22
X	23
Y	24
Z	25
*	26

PARTIE 1 CODAGE

On a programmé en langage Python la fonction crypte qui permet, à l'aide d'un chiffrement affine, de coder une phrase composée des 26 lettres de l'alphabet et du symbole * pour l'espace.
On rappelle que pour effectuer un codage affine d'une phrase, on associe à chaque lettre de l'alphabet un nombre entier comme ci-contre.
On appelle ce codage « chiffrement affine de type $ax + b$ modulo 27 ».

```
alpha="ABCDEFGHIJKLMNOPQRSTUVWXYZ*"

def  crypte(a,b,message):
        code=""
        for caractere in message:
            x = alpha.index(caractere)
            y = (a*x+b)%27
            code = code + alpha[y]
        return code
```

1) a et b sont les paramètres du codage affine.
message est la phrase à coder. Dans la console, il faudra saisir la phrase entre " ".
Expliquer qu'elle est la fonction des instructions suivantes du programme :

- for caractere in message
- x = alpha.index(caractere)
- y = (a*x+b)%27
- code = code + alpha[y]

2) On utilise le chiffrement affine « $4x + 6$ modulo 27 ».
   a) À l'aide du programme, coder la phrase :
      CELUI*QUI*PARLE*TROP*AGIRA*DIFFICILEMENT
   b) Coder une phrase au choix et qui a du sens. N'écrire que la phrase codée.

3) On utilise le chiffrement affine « $5x + 1$ ».
On veut coder la phrase :
C'EST*VERITABLEMENT*S'ENRICHIR*QUE*DE*S'OTER*SES*BESOINS
Modifier le programme pour intégrer le codage d'une apostrophe et donner la phrase codée.

PARTIE 2 DÉCODAGE

Dans la suite, on conserve la possibilité de coder l'apostrophe.
On a programmé en langage Python la fonction « decrypte » qui permet de déterminer la relation de décodage d'une phrase qui a été codée par chiffrement affine. Ce programme est incomplet et sera complété dans la question 2.

```
def decrypte(a,b):
    for i in range(1,28):
        if (a*i)%28==1:
            return ???
```

1) Expliquer l'instruction suivante : `if (a*i)%28==1:`

2) Compléter l'instruction `return` du programme pour qu'il affiche les deux paramètres c et d de la relation de congruence $x \equiv cy + d[28]$ permettant de décoder un message.

3) On considère le chiffrement affine $3x + 6$ modulo 28 ».
À l'aide du programme, trouver la relation de décodage et décoder la phrase suivante :
RUKEARD'SBCHUREAXGEAPSALGAHSBBSAPSARUEAGRMSHBSEARUKEALD SOXBKRHUREAGARUEASRVGRHE

4) Utiliser le programme pour tenter de trouver la relation de décodage pour le chiffrement affine $4x + 6$ modulo 28.
Qu'obtient-on en sortie et pourquoi ?