

Cryptographie :

Le langage Python permet la manipulation de texte (type string ou `str`). On peut alors programmer de nombreuses applications de cryptographie, en voici quelques idées et 2 programmes en Python pour les mettre en œuvre.

- Codage par décalage dit de César : L'idée est que chaque lettre est décalée dans l'alphabet. Pour un décalage de 3 par exemple, les « A » deviennent des « D », les « B » des « E », ..., les « Z » des « C ». Le codage `Rot13` ci-dessous correspond à un tel décalage par 13. Il a l'avantage d'être son propre programme de décodage.

```
def ROT13(texte):
    # implémentation du codage ROT13:
    # décalage par la méthode de César de 13
    l=len(texte)
    code=""
    for i in range(l):
        c=ord(texte[i]) #conversion du caractère
        #en valeur numérique (valeur ASCII)
        c=c+13 # décalage de 13 unités
        if c>122: # gestion du dépassement du z
            c=c-26
        m=chr(c) #reconversion en un caractère
        code=code+m # rajout dans le message codé
    return code
```

- Codage de Vigenère : la clé de chiffrement est un mot, par exemple « ACB ». On décale alors la première lettre de 1 (la valeur numérique du A), la deuxième de 3 et la troisième de 2 puis on recommence 1,3,2,...
- Codage par permutation : chaque lettre de l'alphabet est remplacée par une autre lettre déterminée, cela forme la clé de chiffrement
- Décodage par attaque statistique : pour découvrir la permutation utilisée, on effectue un relevé statistique. En effet en français la répartition des lettres est la suivante :

e = 14,7	u = 6,3	v = 1,6	x = 0,4
s = 7,9	l = 5,5	q = 1,4	y = 0,3
a = 7,6	o = 5,4	f = 1,1	z = 0,1
i = 7,5	d = 3,7	b = 1,0	w = 0,1
t = 7,2	c = 3,3	g = 0,9	k = 0,05
n = 7,1	p = 3,0	h = 0,7	
r = 6,6	m = 3,0	j = 0,5	

Ce programme va ainsi lire un texte en minuscule sans accents et donne les fréquences relevées :

```
texte = open("HugoMiserableCh1.txt", "r")
T=list(texte.read())
l=len(T)

alphabet=['a','b','c','d','e','f','g','h','i','j','k','l','m','n']
frequence=[0]*26

for i in range(26):
    frequence[i]=T.count(alphabet[i])/l
    print(alphabet[i],"-> ", int(frequence[i]*100//l),"pourcent")

texte.close()
```