

10. Codage des caractères

Nous utiliserons dans ce TP le langage de programmation *Python* au moyen de l'environnement de développement *IDLE*.

Les caractères informatiques sont codés par des entiers, il existe différentes normes de codage dont l'ASCII (American Standard Code for Information Interchange) et l'Unicode.

Exemple 1. Codage Ascii en Python : fonctions **chr** et **ord**

```
>>> chr(109)
'm'
>>> ord("n")
110
```

Exercice 1.

- Écrire un programme Python permettant d'afficher la liste des 256 caractères Ascii.
- En déduire les entiers représentant les 26 lettres de l'alphabet en minuscules, les 26 lettres de l'alphabet en majuscules ainsi que les chiffres de 0 à 9.

Exercice 2.

- Écrire une fonction Python **caractere** de paramètre n permettant d'afficher le n -ième caractère de l'alphabet minuscule. (on numérote à partir de 1)
- Écrire une fonction Python **rang** de paramètre c permettant de calculer le rang du caractère c dans l'alphabet minuscule.
- Écrire un programme Python permettant d'écrire l'alphabet dans l'ordre inverse.

Exercice 3.

- Écrire une fonction Python **majuscule** de paramètre c permettant de convertir une chaîne de caractères c de minuscule en majuscule.
- Écrire une fonction Python **minuscule** de paramètre c permettant de convertir une chaîne de caractères c de majuscule en minuscule.

Exercice 4.

Le chiffrement miroir consiste à remplacer chaque caractère alphabétique par celui de même rang dans l'alphabet inversé.

Écrire une fonction Python **miroir** de paramètre c permettant de chiffrer suivant cette méthode la chaîne de caractères c .

Exercice 5. On considère le programme suivant :

```
from tkinter import*
# création de la fenêtre
fenetre=Tk()
fenetre.title('conversion en majuscules')
# creation d'une variable tkinter de type chaîne de caractères
chaine=StringVar()
# création de la zone de saisie de la chaîne de caractères
ZoneDeSaisie=Entry(fenetre,width=100,text=chaine)
ZoneDeSaisie.pack()
# création du bouton de conversion
Bouton=Button(fenetre,text='conversion')
Bouton.pack()
# définition du gestionnaire de l'évènement click
def conversion(self):
    l=chaine.get()
    L=""
    for k in range(0,len(l)):
        if 97<=ord(l[k])<=122:
            L=L+chr(ord(l[k])-32)
        else:
            L=L+l[k]
    chaine.set(L)
Bouton.bind('<ButtonPress-1>',conversion)
# lancement de la boucle du programme
fenetre.mainloop()
```

- Tester ce programme.
- Analyser le fonctionnement du programme et le modifier pour qu'il réalise le chiffrement miroir.

Réponses

- 1)

```
for k in range(0,256):
    print(k, chr(k))
```

Alphabet minuscule : de 97 à 122
Alphabet majuscule : de 65 à 90
Chiffres : de 48 à 57
- 2)

```
def caractere(n):
    '''affichage du n-ième caractère de l'alphabet minuscule'''
    print(chr(96+n))
def rang(c):
    '''calcul du rang du caractère c dans l'alphabet minuscule'''
    return(ord(c)-96)
for k in range(1,27):
    print(chr(123-k))
```
- 3)

```
def majuscule(c):
    '''conversion de la chaîne de caractères c de minuscule en majuscule'''
    C=""
    for k in range(0,len(c)):
        if 97<=ord(c[k])<=122:
            C=C+chr(ord(c[k])-32)
        else:
            C=C+c[k]
    return(C)
def minuscule(c):
    '''conversion de la chaîne de caractères c de majuscule en minuscule'''
    C=""
    for k in range(0,len(c)):
        if 65<=ord(c[k])<=90:
            C=C+chr(ord(c[k])+32)
        else:
            C=C+c[k]
    return(C)
```
- 4)

```
def miroir(c):
    '''chiffrement miroir de la chaîne de caractères c'''
    C=""
    for k in range(0,len(c)):
        if 97<=ord(c[k])<=122:
            C=C+chr(219-ord(c[k]))
        else:
            if 65<=ord(c[k])<=90:
                C=C+chr(155-ord(c[k]))
            else:
                C=C+c[k]
    return(C)
```

```
5) from tkinter import*
# création de la fenêtre
fenetre=Tk()
fenetre.title('chiffrement miroir')
# creation d'une variable tkinter de type chaîne de caractères
chaine=StringVar()
# création de la zone de saisie de la chaîne de caractères
ZoneDeSaisie=Entry(fenetre,width=100,text=chaine)
ZoneDeSaisie.pack()
# création du bouton de chiffrement
Bouton=Button(fenetre,text='chiffrement')
Bouton.pack()
# définition du chiffrement miroir
def miroir(c):
    '''chiffrement miroir de la chaîne de caractères c'''
    C=""
    for k in range(0,len(c)):
        if 97<=ord(c[k])<=122:
            C=C+chr(219-ord(c[k]))
        else:
            if 65<=ord(c[k])<=90:
                C=C+chr(155-ord(c[k]))
            else:
                C=C+c[k]
    return(C)
# définition du gestionnaire de l'évènement click
def chiffrement(self):
    l=chaine.get()
    chaine.set(miroir(l))
Bouton.bind('<ButtonPress-1>',chiffrement)
# lancement de la boucle du programme
fenetre.mainloop()
```