

Exercices Corrigés Python (série 6)

Exercice 1

Un administrateur d'un site web veut assurer un maximum de sécurité pour les utilisateurs du site. Pour ceci il décide de réaliser une application qui évalue la force des mots de passe des différents utilisateurs du site, sachant qu'un mot de passe est une chaîne de caractères qui ne comporte pas d'espaces et de lettres accentuées.

La force d'un mot de passe varie, selon la valeur d'un score calculé, de 'Très faible' jusqu'à 'Très fort' :

- Si le score <20 , la force du mot de passe est 'Très faible'
- Sinon si le score <40 , la force d'un mot de passe est 'Faible'
- Sinon si le score <80 , la force du mot de passe est 'Fort'
- Sinon la force du mot de passe est 'Très fort'

Le score se calcule en additionnant des bonus et en retranchant des pénalités.

Les bonus attribués sont :

- Nombre total de caractères * 4
- (Nombre total de caractères – nombre de lettres majuscules) * 2
- (Nombre total de caractères – nombre de lettres minuscules) * 3
- Nombre de caractères non alphabétiques * 5

Les pénalités imposées sont :

- La longueur de la plus longue séquence de lettres minuscules * 2
- La longueur de la plus longue séquence de lettres majuscules * 3

Exemple :

Pour le mot de passe 'P@cSI_promo2017', le score se calcule comme suit :

La somme de bous = $15*4 + (15-3) *2 + (15-6) *3+6*5=141$

- Le nombre total de caractères = 15
- Le nombre de lettres majuscules = 3
- Le nombre de lettres minuscules=6
- Le nombre de caractères non alphabétiques =6

La somme des pénalités = $5*2+2*2=14$

- La longueur de la plus longue séquence de lettres minuscules('promo')=5
- La longueur de la plus longue séquence de lettres majuscules('SI')=2

Le score final = $141-14=127$; puisque $127>80$ alors le mot de passe est considéré comme 'Très fort'

Travail demandé :

1. Ecrire une fonction **NbCMin(pass)** qui retourne le nombre de caractères minuscules.
2. Ecrire une fonction **NbCMaj(pass)** qui retourne le nombre de caractères majuscules.
3. Ecrire une fonction **NbCAlphapass)** qui retourne le nombre de caractères non alphabétiques.
4. Ecrire une fonction **LongMaj(pass)** retourne la longueur de la plus longue séquence de lettres majuscules.
5. Ecrire une fonction **LongMin(pass)** retourne la longueur de la plus longue séquence de lettres minuscules.
6. Ecrire une fonction **Score(pass)** qui affiche le score d'un mot de passe

Correction :

```
def NbcMin(passe):
    nb=0
    for i in passe:
        if 'a'<=i<='z':
            nb+=1
    return nb

def NbcMaj(passe):
    nb=0
    for i in passe:
        if 'A'<=i<='Z':
            nb+=1
    return nb

def NbcAlpha(passe):
    return len(passe)-NbcMaj(passe)-NbcMin(passe)

def longMaj(passe):
    d=0
    s=0
    i=0
    while i< len(passe):
        if 'A'<passe[i]<'Z':
            s+=1
        else:
            if s > d:
                d=s
                s=0
        i+=1
    return d

def longMin(passe):
    d=0
    s=0
    i=0
    while i< len(passe):
        if 'a'<passe[i]<'z':
```

```

        s+=1
    else:
        if s > d:
            d=s
            s=0
        i+=1
    return d

```

```

def score(password):
    bonus=(len(password)-NbcMin(password))*3+(len(password)-
NbcMaj(password))*2+(len(password)-NbcAlpha(password))*5
    penalites=longMaj(password)*3+longMin(password)*2
    val=bonus-penalites
    if val<20:
        print('Très faible')
    elif val<40:
        print('Faible')
    elif val<80:
        print('Fort')
    else:
        print('Très fort')
pas="P@SI_promo2016"
score(pas)

```

Exercice 2 :

Louis Braille, est l'inventeur du système d'écriture tactile à points saillants, à l'usage des personnes aveugles ou fortement malvoyantes.

En braille standard :

- Un caractère est représenté par six points numérotés de 1 à 6 et disposés comme le montre la **Figure 1**
- Un point peut être saillant (en relief) ou non, comme le montre la **Figure 2**.
- Le nombre et la disposition des points en relief définissent un caractère.

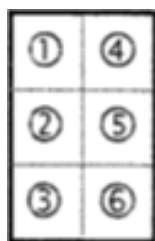


Figure 1



Figure 2


```
****_**_*_*_
```

Le programme affichera la chaîne : 'BONJOUR PSI'

En effet :

- 'BONJOUR' est l'équivalent en alphabet français de la première ligne du fichier 'Braille.txt'.

```

    B         O         N         J         O         U         R
**_**_*_*_  *_**_*_*_  *_**_*_*_  *_**_*_*_  *_**_*_*_  *_**_*_*_

```

- 'PSI' est l'équivalent en alphabet français de la deuxième ligne du fichier 'braille.txt'

Ecrire un programme Python permettant de résoudre ce problème. Vous pouvez utiliser plusieurs fonctions

Correction :

```

def dictionnaire(fichier) :
    fichier=open(fichier)
    dic={}
    for i in code:
        c=i.strip()
        l=c.split(' ')
        cle=l[1]
        dic[cle]=l[0]
    code.close()
    return dic

def coder(fichier) :
    dic=dictionnaire('codes_Braille.txt')
    f=open(fichier)
    s=""
    for j in f:
        c=j.strip()
        nb=len(j)//6
        po=0
        for k in range(nb):
            cle=c[po:po+6]
            s+=dic[cle]
            po+=6
        s+=' '
    return s

print(coder('braille.txt'))

```