

# Sujet 5

## EXERCICE 1 (10 points)

Le nombre d'occurrences d'un caractere dans une chaine de caractere est le nombre d'apparitions de ce caractere dans la chaine.

Exemples :

- le nombre d'occurrences du caractere 'o' dans 'bonjour' est 2 ;
- le nombre d'occurrences du caractere 'b' dans 'Bebe' est 1 ;
- le nombre d'occurrences du caractere 'B' dans 'Bebe' est 1 ;
- le nombre d'occurrences du caractere ' ' dans 'Hello world' est 2.

On cherche les occurrences des caracteres dans une phrase. On souhaite stocker ces occurrences dans un dictionnaire dont les clefs seraient les caracteres de la phrase et les valeurs [l'occurrence de ces caracteres.

Par exemple : avec la phrase 'Hello world' le dictionnaire est le suivant :

{ 'H' : 1, 'e' : 1, 'l' : 3, 'o' : 2, ' ' : 2, 'w' : 1, 'r' : 1, 'd' : 1, '!' : 1 }

*L'ordre des clefs n'a pas d'importance.*

Ecrire une fonction `nbr_occurrences` prenant comme parametre une chaine de caracteres `chaine` et renvoyant le dictionnaire des nombres d'occurrences des caracteres de cette chaine.

## EXERCICE 2 (10 points)

On s'intéresse dans cet exercice à la recherche dichotomique dans un tableau trié d'entiers.

Compléter la fonction suivante en respectant la spécification.

```
def dichotomie(tab, x):  
    """  
    tob : tableau d'entiers trié dans l'ordre  
    croissant x : nombre entier  
    La fonction renvoie True si tab contient x et False sinon  
    """  
    debut = 0  
    fin = len(tab) - 1  
    while debut <= fin:  
        if x == tab[m]:  
            return ..  
        . if x >  
        tab[m]:  
            debut = m + 1  
        else:  
            fin  
    = ...  
    return ...
```

Exemples:

```
>>> dichotomie([15, 16, 18, 19, 23, 24, 28, 29, 31, 33],28)
```

True

```
>>> dichotomie([15, 16, 18, 19, 23, 24, 28, 29, 31, 33],27)
```

False