

```
# -*- coding: utf-8 -*-  
"""
```

```
Created on Sun Jan 12 14:58:04 2014
```

```
@author: dconduche  
"""
```

```
def mot_est_dans_texte(mot, texte):  
    """Recherche la présence d'une chaîne de caractères dans un texte.  
  
    mot -- le motif (string) à rechercher, qui peut contenir plusieurs mots  
    texte -- le texte (string) où l'on cherche "mot"  
    """  
    n = len(mot)  
    for i in range(len(texte)-n+1):  
        if texte[i:i+n] == mot:  
            return True  
    return False
```

```
def indice_mot_dans_texte(mot, texte):  
    """Recherche l'indice de début d'un mot dans un texte.  
  
    Keyword arguments:  
    mot -- le motif (string) à rechercher, qui peut contenir plusieurs mots  
    texte -- le texte (string) où l'on cherche "mot"  
    retourne le plus grand indice si le motif est présent, et -1 sinon.  
    """  
    n = len(mot)  
    indice = -1  
    for i in range(len(texte)-n+1):  
        if texte[i:i+n] == mot:  
            indice = i  
    return indice
```

```
#Tests :  
TexteTest = 'Ce test est un petit test'  
print(TexteTest)  
M = ['Ceci', 't u', 'test', 'blabla']  
# Liste de motifs à chercher. Présents ou non, au bout ou non.
```

```
for mot in M:  
    print('Présence de \'' + mot + '\'' dans le texte :'  
          + str(mot_est_dans_texte(mot, TexteTest)))  
  
for mot in M:  
    print('Indice de \'' + mot + '\'' dans le texte :'  
          + str(indice_mot_dans_texte(mot, TexteTest)))
```

```
"""On vous demandera la première fonction.  
Il ne faut pas que cela vous empêche de comprendre la seconde, ni même  
de l'apprendre.  
Vous n'avez pas à écrire ce qu'il y a entre triple quotes (docstring)  
Ni les tests bien sûr.  
Vous pouvez avoir votre propre version (nom des variables, etc) : l'essentiel  
c'est que l'algorithme fonctionne (en cas de doute, demandez-nous).  
"""
```