

# Corrigé sujet 11 - Année : 2022

[Sujet 11 - 2022](#) ↴

## 1. Exercice 1



```

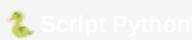
1 def recherche(tab,elt):
2     ind_debut = 0
3     ind_fin = len(tab)-1
4     while ind_fin > ind_debut:
5         ind_milieu = (ind_fin+ind_debut)//2
6         if tab[ind_milieu]==elt:
7             return ind_milieu
8         elif tab[ind_milieu]>elt:
9             ind_fin=ind_milieu-1
10        else:
11            ind_debut=ind_milieu+1
12    return -1

```

### Commentaire

- Bien qu'au programme (méthode diviser pour régner), cet exercice est bien plus difficile que ceux proposés habituellement en exercice 1 (recherche simple, recherche de maximum, calcul de moyennes, ... ).
- En cas de difficultés, revenir au chapitre de première sur la [recherche par dichotomie](#)

## 2. Exercice 2



```

1 ALPHABET='ABCDEFGHIJKLMNPQRSTUVWXYZ'
2
3 def position_alphabet(lettre):
4     return ALPHABET.find(lettre)
5
6 def cesar(message, decalage):
7     resultat = ''
8     for lettre in message : #(1)
9         if lettre in ALPHABET :
10             indice = (position_alphabet(lettre) + decalage)%26 #(2)
11             resultat = resultat + ALPHABET[indice]

```

```
12     else:  
13         resultat = resultat + lettre #(3)  
14     return resultat
```

1. La variable de parcours du message s'appelle `lettre` comme indiqué à la ligne suivante !
2. On ajoute le decalage à la position de la lettre le `%26` sert ensuite à s'assurer qu'on retombe entre 0 et 25.
3. Si la lettre n'est pas dans l'alphabet, on la laisse telle quelle

### Attention

1. La variable `lettre` de parcours du message peut contenir justement autre chose qu'une lettre (ponctuation, espace, ...)
2. La méthode `find` de recherche de l'indice de la lettre dans l'alphabet pourrait s'écrire sans utiliser la méthode `find` (par exemple avec `ord`)