

Exercice 1 5 points

lst = [3, 2, 4, 1, 5].

1.
 - a. afficher le 4ème élément de la liste : `lst[3]`
 - b. afficher la longueur de la liste : `len(lst)`
 - c. que le 2ème élément devienne égal à 8 : `lst[1] = 8`
2.
 - a. `lst[-1] : 5`
 - b. `lst[5] : Error: list index out of range`

Exercice 2 4 points

On souhaite afficher un par un les éléments de la liste `lst`.

1. Écrire un code qui utilisera obligatoirement un parcours **par élément**.

```
lst = ["janvier", "février", "mars", "avril", "mai"]
for mois in lst:
    print(mois)
```

2. Écrire un code qui utilisera obligatoirement un parcours **par indice**.

```
lst = ["janvier", "février", "mars", "avril", "mai"]
for i in range(len(lst)):
    print(lst[i])
```

Exercice 3 3 points

```
for i in range(len(tab)-1):
    if tab[i] == tab[i+1]:
        print(tab[i], tab[i+1])
```

Exercice 4 4 points

1. Créer une liste `lst` contenant 100 fois le nombre 0. (la méthode de construction est libre)

```
lst = [0]*100
# ou bien
lst = []
for _ in range(100):
    lst.append(0)
```

2. Remplacer chaque élément de `lst` par le mot 'pair' ou 'impair' suivant que l'indice de l'élément est pair ou impair.

```
for i in range(100):
    if i % 2 == 0:
        lst[i] = 'pair'
    else:
        lst[i] = 'impair'
```

Exercice 5 4 points

On considère la liste

tab = ["un", "deux", "trois", "quatre", "cinq", "six", "sept", "huit", "neuf"]

1. Écrire une fonction `nb_voyelles` qui prend en paramètres une chaîne de caractères `mot` et qui renvoie le nombre de voyelles de `mot`.

```
def nb_voyelles(mot):
    voyelles = "aeiouy"
    n = 0
    for l in mot:
        if l in voyelles:
            n += 1
    return n
```

2. Écrire un code permettant de construire une liste `new_tab` qui contient les éléments de `tab` ayant exactement 2 voyelles.
(Vous pouvez vous servir de la question précédente même si vous ne l'avez pas traitée.)

```
new_tab = []
for mot in tab:
    if nb_voyelles(mot) == 2:
        new_tab.append(mot)
```