

### EXERCICE 1 - ●○○○ - LISTE INVERSÉE

Écrire une fonction qui prend en argument d'entrée une liste et qui renvoie la liste obtenue en inversant l'ordre des éléments.

```
1  
2  
3  
4  
5
```

#### Petite remarque

La commande `L[::-1]` permet d'obtenir la liste inversée, mais il s'agit ici de ne pas l'utiliser.

### EXERCICE 2 - ●○○○ - SUPPRESSION

Écrire une fonction qui prend en arguments d'entrée une liste et un élément  $x$ , puis qui renvoie la nouvelle liste obtenue après suppression de toutes les occurrences de  $x$  dans cette liste.

```
1  
2  
3  
4  
5  
6  
7  
8
```

#### Petite remarque

Le programme "naïf" consistant à supprimer `L[k]` lorsqu'il est égal à  $x$  pose souci... Et autant profiter des spécificités de Python !

### EXERCICE 3 - ●○○○ - INSERTION

Écrire une fonction qui prend en arguments d'entrées une liste, un indice  $i$  et un élément  $x$ , puis qui renvoie la nouvelle liste obtenue après insertion de l'élément  $x$  à l'indice  $i$ , en décalant le reste vers la droite.

```
1  
2  
3  
4  
5
```

#### Petite remarque

La commande `L.insert(i,x)` permet de le faire, mais elle n'est pas à connaître.

### EXERCICE 4 - ●○○○ OCCURRENCES...

1. Écrire une fonction prenant en arguments d'entrée une liste  $L$  et un élément  $x$ , puis renvoyant en sortie le nombre d'occurrences de  $x$  dans  $L$ .

```
1  
2  
3  
4  
5  
6
```

2. Écrire une fonction prenant en arguments d'entrée une liste  $L$  et un élément  $x$ , puis renvoyant en sortie le rang de la première occurrence de  $x$  dans  $L$  si la liste  $L$  contient l'élément  $x$ , et `False` sinon.

```
1  
2  
3  
4  
5
```

### EXERCICE 5 - ●○○○ MAXIMUM

1. Sans utiliser la commande `max`, écrire une fonction prenant en argument d'entrée une liste de réels `L` puis renvoyant en sortie le maximum de `L` ainsi que son nombre d'occurrences.

```
1
2
3
4
5
6
7
8
9
10
```

2. Sans utiliser la commande `max`, écrire une fonction prenant en argument d'entrée une liste de réels `L` puis renvoyant en sortie le maximum de `L` ainsi que le rang de sa première occurrence.

```
1
2
3
4
5
6
7
8
```

3. Sans utiliser la commande `max`, écrire une fonction prenant en argument d'entrée une liste de réels `L` puis renvoyant en sortie les deux plus grands nombres (éventuellement égaux) de cette liste.

```
1
2
3
4
5
6
7
8
9
10
11
```

4. Sans utiliser la commande `max`, écrire une fonction prenant en argument d'entrée une liste de réels `L` dont au moins deux sont distincts puis renvoyant en sortie le second plus grand maximum (distinct du premier) de cette liste.

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
```

### EXERCICE 6 – ●●○○ LES PLUS PROCHES

Écrire une fonction prenant en argument d'entrée une liste de réels `L` (contenant au moins deux éléments) puis renvoyant en sortie les deux éléments les plus proches.

1  
2  
3  
4  
5  
6  
7  
8  
9

**Petite remarque**

En mettant  
◀ **if 0<abs(L[i]-L[j])<ecart**  
en ligne 6, on obtient les deux  
plus proches distincts.

**EXERCICE 7 - ●●●○ - ORDONNER UNE LISTE...**

Écrire une fonction qui prend en arguments d'entrée une liste de réels et qui renvoie la liste triée dans l'ordre croissant.

1  
2  
3  
4  
5  
6

**Petite remarque**

◀ La commande **L.sort()** permet  
d'ordonner une liste, mais il s'agit  
ici de ne pas l'utiliser.