

Exercice 1 : On se donne une liste de nombres *maListe*.

Ecrire une fonction *est_triee(maListe)* permettant de vérifier si *maListe* est ordonnée de manière croissante.

Exercice 2 : On considère la liste A=[3,7,1,2,5]

- 1) Applique le tri par sélection à la liste A en indiquant les étapes.
- 2) Applique le tri par insertion à la liste A en indiquant les étapes.

Exercice 3 : Teste la fonction suivante :

```
from random import randint
def fonctionInconnue(n):
    ml=[i for i in range(n)]
    r=[]
    for i in range(n):
        e=randint(0,len(ml)-1)
        r.append(ml[e])
        ml.remove(ml[e])
    return r
```

Une fois son utilisation déterminée, renomme la fonction et documente-la en y mettant des exemples.

Exercice 4 : Utilise ici le code du tri par sélection.

- 1) Ecrit le code en y insérant la documentation et un exemple
- 2) Ecrit une autre fonction de ce tri par sélection mais cette fois-ci donnant la liste triée par ordre décroissant.

Le coût a-t-il changé ?

Exercice 5 : Utilise ici le code du tri par insertion.

- 1) Ecrit le code en y insérant la documentation et un exemple
- 2) Ecrit une autre fonction de ce tri par insertion mais cette fois-ci donnant la liste triée par ordre décroissant.

Le coût est-il inchangé dans tous les cas ?

Exercice 6 : un autre, le **tri à bulle**

- 1) Voici les étapes de ce tri à la liste [3,2,7,1,5] :

```
[3,2,7,1,5]
[2,3,7,1,5]
[2,3,7,1,5]
[2,3,1,7,5]
[2,3,1,5,7]
[2,3,1,5,7]
[2,1,3,5,7]
[2,1,3,5,7]
[1,2,3,5,7]
[1,2,3,5,7]
[1,2,3,5,7]
```

Combien d'étapes sont proposées ?

- 2) Décrire la méthode employée afin de faire apparaître l'algorithme.
- 3) Ecrire l'algorithme en Python et montrer que le coût est quadratique.