

NSIT	e07a	Révisions	
Kata 1			
Tests unitaires, spécifications, listes, récursivité			1/1

**Exercice** : Trouver les deux extrêmes de la liste (plus petit et plus grand éléments)

**Énoncé** : C'est écrit dans le titre ;)

```
# On peut s'aider de la fonction tri_selection() vu précédemment
def extremum(ma_liste:list)->tuple:
    """ Renvoie l'élément le plus petit et le plus grand sous la forme du p-uplet (minimum, maximum)
        Précondition: ma_liste contient au moins deux éléments

    :param list ma_liste: Liste à examiner.

    :return: Le couple constitué de (minimum, maximum)

    >>> extremum(['A','R','Z'])
    ('A', 'Z')
    """
    assert len(ma_liste) >= 2, 'La liste doit contenir au moins 2 éléments'
    ma_liste_copie = tri_selection(ma_liste.copy())
    ...
    return (minimum, maximum)

import doctest
doctest.testmod(verbose=True)
L = [randint(1,100) for i in range(1,20)] # Tire une liste aléatoire
print(L)
extremum(L)
```

### Commentaire :

Pour récupérer le dernier élément d'une liste (une\_liste), il existe deux méthodes :

1- En retrouvant le dernier indice de la liste

```
dernier_indice = len(une_liste) - 1
une_liste[dernier_indice]
```

2- Plus rapide

```
une_liste[-1]
```

Les indices négatifs permettent en effet de parcourir une liste dans le sens inverse en commençant par le dernier éléments

valeurs	20	-4	3	99	1	8	-30
indice	0	1	2	3	4	5	6
indice	-7	-6	-5	-4	-3	-2	-1

Ainsi, une\_liste[5] et une\_liste[-2] renvoient tous les deux la valeur 8

N'oubliez pas de compléter le docstring avec les tests demandés :

**Complétez le docstring avec 3 tests sur une liste d'entiers positifs, négatifs et de nombres décimaux. (cf activité 02b)**