

# Correction

NSI - 2021 Métropole Libre Jour 2 (21-NSIJ2ME2)

## Exercice 4 - Mélange des éléments d'une liste

1. Ce code ne réalise pas un échange, il écrase la valeur d'indice `i2` par la valeur d'indice `i1`. Ainsi la valeur initiale d'indice `i2` est perdue.

**Solution 1 :**

```
def echange(lst, i1, i2):  
    a = lst[i1]  
    lst[i1] = lst[i2]  
    lst[i2] = a
```

**Solution 2 : Affectation multiple :**

```
def echange(lst, i1, i2):  
    lst[i2], lst[i1] = lst[i1], lst[i2]
```

2. Les valeurs qui peuvent être renvoyée par la fonction `randint(0, 10)` sont : 0, 1, 9 et 10.

3. a. La fonction mélange se termine car à chaque appel récursif, la valeur `ind` diminue, ce qui rendra fausse la condition `ind > 0` et arrêtera l'empilement d'appel récursif.

3. b. Pour une liste de longueur  $n$ , il y a  $n - 2$  appels récursifs. En effet, on ne compte pas le premier appel (pour la première valeur de  $n$ ) et pour la dernière valeur de  $n$  ( $n = 0$ ) il n'y a pas d'appel.

3. c. Tableau d'exécution :

Lst (affichage)	ind	j
[0, 1, 2, 3, 4]	4	2
[0, 1, 4, 3, 2]	3	1
[0, 3, 4, 1, 2]	2	2
[0, 3, 4, 1, 2]	1	0
[3, 0, 4, 1, 2]	0	

3. d.

**Solution 1 :**

```
def melange_it(lst, ind):  
    for i in range(ind):  
        j = randint(0, ind)  
        echange(lst, ind-i, j)
```

**Solution 2 :**

```
def melange_it(lst, ind):  
    for i in range(ind, 0, -1):  
        j = randint(0, i)  
        echange(lst, i, j)
```