

Exercice n° 1 Schuffle

Soit la liste $L=[0,1,2,3,4,5,6,7,8,9]$

1. Compléter la fonction ci-dessous :

```
def permuterV1(i,j) :  
    """  
    Permute les éléments i et j de la liste L  
    param i, j : (int)  
    return L : (list)  
    C.U :  $i \leq 9, j \leq 9$  et la liste L doit avoir été créée.  
    Exemple :  
    >>> permuter(2,7)  
    [0,1,7,3,4,5,6,2,8,9]  
    """  
  
    return L
```

2. Compléter la fonction ci-dessous :

```
def ChoisirV1() :  
    """  
    Choisir aléatoirement deux entiers entre 0 et 9.  
    param : None (Aucun)  
    return ListeAlea : (list) une liste contenant deux entiers aléatoires.  
    Exemple :  
    >>> Choisir()  
    [6,9]  
    """  
  
    return ListeAlea
```

3. Compléter la fonction ci-dessous :

```
def ChoisirV2() :  
    """  
    Choisir aléatoirement deux entiers différents entre 0 et 9.  
    param : None (Aucun)  
    return ListeAlea : (list) une liste contenant deux entiers aléatoires différents.  
    Exemple :  
    >>> Choisir()  
    [9,3]  
    """  
  
    return ListeAlea
```

4. Écrire une fonction **permuterV2()** qui ne prend pas de paramètre et qui renvoie L après avoir permuté deux éléments de L choisis au hasard. (N'oubliez pas d'écrire la documentation)
5. Executer 1000 fois la fonction **permuterV2()**.
6. Quel était l'objectif de l'exercice ?
7. **- BONUS -** Compléter la fonction ci-dessous :

```
def melanger(liste) :  
    """  
    Mélange les éléments de liste.  
    param liste : (list)  
    return liste : (list) liste dont les éléments ont été mélangé.  
    """  
  
    return liste
```

Exercice n° 2 Nombre palindrome

Définition : Un nombre palindrome est un entier positif qui peut se lire dans les deux sens, c'est-à-dire que la séquence de ses chiffres est la même lorsqu'il est lu de droite à gauche ou de gauche à droite.

Par exemple : 66, 161, 1457541.

Considérons l'algorithme suivant :

Prenons un nombre entiers nb (qui n'est pas forcément un palindrome).

Par exemple $nb = 1473$.

1^{ère} étape : On écrit nb à l'envers, on obtient $nbEnvers = 3741$.

2^{ème} étape : On additionne nb et $nbEnvers$, on obtient 5214.

3^{ème} étape : On stocke le résultat dans nb . $nb = 5214$.

On répète les trois étapes ci-dessus tant que nb n'est pas un palindrome.

Exemples :

Avec $nb = 1473$, on a $1473 + 3761 = 5214$ puis recommence $5214 + 4125 = 9339$ c'est un palindrome.

Avec $nb = 28$, on a $28 + 82 = 110$, on recommence $110 + 011 = 121$, c'est un palindrome.

1. Appliquer l'algorithme avec 1969 puis 79.
2. A l'aide de votre cours, retrouver la fonction **palindrome(chaine)** qui teste si une chaîne de caractère est un palindrome.
3. Tester **palindrome("168861")** puis **palindrome("20092009")**.
4. Ecrire une fonction **NbPalindrome(nb)** qui prend en paramètre un entier et qui après avoir effectué l'algorithme ci-dessus, renvoie la liste des valeurs de nb.

Aide :

```
>>> str(1515)
>>> int("1515")
>>> a=str(1515)
>>> a[: -1]
>>> type(str(1515))
>>> type(int("1515"))
>>> type(1515)
>>> type("1515")
```

5. a. Vérifier le bon fonctionnement de votre script.

```
>>>NbPalindrome(1789)
[1789, 11660, 18271, 35552, 61105, 111221, 233332]
>>>NbPalindrome(89)
[89, 187, 968, 1837, 9218, 17347, 91718, 173437, 907808, 1716517, 8872688, 17735476, 85189247, 159487405,
664272356, 1317544822, 3602001953, 7193004016, 13297007933, 47267087164, 93445163438, 176881317877,
955594506548, 1801200002107, 8813200023188]
```

- b. Tester l'algorithme avec $nb = 196$.
- c. Lire la page wikipedia : Nombre de Lychrel
- d. Proposer une amélioration de la fonction **NbPalindrome**.