

---

## TP1 Python

### Prérequis :

Liste, test, boucles for et while, fonctions range, randint, print

#### Exercice 1

Un professeur dispose d'une liste Python  $L$  constituée des notes de ses élèves à un contrôle.

- 1) Quelle commande permet d'afficher  $L$  ? la moins bonne note ? la meilleure note ?
- 2) Le professeur décide de mettre 1 point de plus aux élèves qui n'ont pas eu la moyenne au contrôle. Écrire un programme qui affiche à l'écran la nouvelle liste  $L$  ainsi formée et l'exécuter avec la liste  $L$  de votre choix.

#### Exercice 2

On considère la liste intitulée nombres et définie par :

```
nombres=[k for k in range(5,100,3)]
```

- 1) Quelles valeurs contient la liste nombres ?
- 2) Pour chacun des cas suivants, écrire une commande réalisant les actions demandées :
  - (i) afficher la liste,
  - (ii) ajouter l'élément 40 à la fin de la liste,
  - (iii) afficher le cinquième élément de la liste,
  - (iv) afficher la sous-liste du 5ème au 10ème élément,
  - (v) afficher le dernier élément de la liste,
  - (vi) supprimer le septième élément de la liste.

#### Exercice 3

On dispose d'une liste  $L$  d'entiers et du programme suivant :

```
occ=0
for k in range(len(L)):
    if L[k]==4:
        occ=occ+1
print(occ)
```

- 1) a) Que signifie  $\text{len}(L)$  ? Quelles sont les valeurs prises par  $k$  ?  
b) Que renvoie ce programme ?
- 2) Avec quelle commande très simple pouvait-on obtenir le même résultat que le programme ?
- 3) Exécutez le programme avec la liste  $L$  de votre choix.

---

#### Exercice 4

On dispose d'une liste  $L$  d'entiers et du programme suivant :

```
pos=[]
for k in range(len(L)):
    if L[k]==7:
        pos.append(k)
print(pos)
```

- 1) Expliquez ce que renvoie ce programme.
- 2) Exécutez-le en prenant  $L = [1,3,8,7,4,5,6,2,7]$ , puis  $L = [5,4,6,3,1,2]$ .

#### Exercice 5

- 1) Que va afficher le programme suivant ?

```
L=[]
for i in range(700,1000):
    if (i%7==0)and(i%2!=0):
        L.append(i)
print(L)
print("Le nombre cherché est :",len(L))
```

- 2) Exécutez-le.

#### Exercice 6

On considère le programme suivant :

```
import numpy.random as rd
liste=[rd.randint(1,11) for k in range(5)]
print(liste)
```

- 1) a) Expliquez la signification du mot `rd`.  
Que désigne `rd.randint(1,11)` ?
- 1) b) Que renvoie ce programme ?
- 2) Tom complète ce programme par les commandes ci-dessous :

```
for i in range(5):
    if liste[i]==10:
        del liste[i]
print(liste)
```

- a) Tom exécute le programme en entier plusieurs fois de suite et ne comprend pas pourquoi de temps en temps, la console lui renvoie le message d'erreur : « *IndexError : list index out of range* ». Exécutez le programme, puis expliquez à Tom son erreur.
- b) Que voulait faire Tom ? Rectifiez le programme.

---

### Exercice 7

On considère la fonction mystère dont l'argument est une liste.

```
def mystere(old_list):
    idx=len(old_list)-1
    new_list=[]
    while idx>=0:
        new_list.append(old_list[idx])
        idx=idx-1
    return new_list
```

- 1) Quel est le but de cette fonction ?
- 2) On prend `old_list=[2,5,3,7,9]`. Que retourne la fonction ?

### Exercice 8

- 1) Compléter les pointillés du programme afin qu'il affiche le plus petit élément et le plus grand élément de la liste  $L$ .

```
max=.....
min=.....
for i in range(len(L)):
    if L[i] ..... max:
        max=L[i]
    if L[i] ..... min:
        min=L[i]
print(max, '\n', min)
```

- 2) On pouvait obtenir la même chose en utilisant deux commandes très simples. Lesquelles ?
- 3) Exécutez le programme avec la liste  $L$  de votre choix.

### Exercice 9

On considère la liste intitulée `ma_liste` donnée par :

```
ma_liste=[11,2,3,5,3,7,15,8,2,4]
```

La commande `ma_liste.index(x)` renvoie le premier indice pour lequel la valeur de `ma_liste` fait  $x$ .

Par exemple, `ma_liste.index(15)` renvoie 6.

Que va afficher le programme suivant ?

```
ma_liste=[11,2,3,5,3,7,15,8,2,4]
new_liste=[]
while len(ma_liste)>0:
    new_liste.append(min(ma_liste))
    idx=ma_liste.index(min(ma_liste))
    del ma_liste[idx]
print(new_liste)
```