

PLANCHE 5 SANS PREPARATION**Informatique**

On s'intéresse dans cet exercice à des listes d'entiers. L'utilisation de `max` ou de `count` est interdite.

- ① Écrire une fonction prenant en argument une liste d'entiers `L` et un entier `k`. Cette fonction renvoie `True` si tous les entiers de cette liste sont compris entre 0 et `k` et `False` sinon.

Par exemple : Pour `L=[0, 2, 0]` la fonction renvoie `True` si `k = 2` ou `k = 3` et `False` si `k = 1`.

- ② Écrire une fonction de mêmes arguments, où la liste est supposée ne contenir que des entiers compris entre 0 et `k`. Cette fonction renvoie l'élément le plus fréquent (ou l'un d'entre eux s'il y en a plusieurs).

Par exemple : Pour `L=[0, 4, 0, 1, 4]` et `k = 5` la fonction renvoie 0 ou 4.

Correction :

```
1 def verif(L, k):
2     n = len(L)
3     for i in range(n):
4         if L[i]<0 or L[i]>k:
5             return False
6     return True
```

```
1 def plusFrequent(L, k):
2     assert verif(L, k) == True
3     n = len(L)
4     F = [0]*(k+1)
5     for i in range(n):
6         F[L[i]] += 1
7     el_plus_f, m = 0, 0
8     for j in range(k+1):
9         if F[j] > m:
10             el_plus_f = j
11             m = F[j]
12     return el_plus_f
```

Autre méthode par les dictionnaires :

```
1 def dico(L, k):
2     dico = {}
3     for c in L:
4         if c in dico:
5             dico[c] += 1
6         else:
7             dico[c] = 1
8     return dico
9
10 def dico2(L, k):
11     D = {i:0 for i in range(k+1)}
12     for i in range(len(L)):
13         D[L[i]] += 1
14     return D
15
16 def plusFrequentDico(L, k):
17     D = dico(L, k)
18     el_plus_f = 0
19     fM = 0
20     for cle, valeur in D.items():
21         if valeur > fM:
22             fM = valeur
23             el_plus_f = cle
24     return el_plus_f
```