

NUMERIQUE et SCIENCES INFORMATIQUES

Épreuve de l'enseignement de spécialité

Sujet d'entraînement

Partie Pratique

Classe Terminale de la voie générale

Le candidat doit traiter les 2 exercices

Le sujet comporte 3 pages



Exercice 1 (4 points)

Vous gérez un site de vente en ligne.

Vous stockez l'ensemble des ventes successives réalisées une heure donnée dans une liste `ventes`, liste qui contient des nombres entiers. Chaque nombre entier fait référence à un article vendu.

Afin de gérer ce site, vous devez créer une fonction `comptage` qui prend en paramètre cette liste `ventes` et renvoie le dictionnaire où chaque nombre entier référençant un article vendu est une clé et où le nombre de fois où cet article a été vendu dans l'heure est la valeur associée à cette clé. De plus, la clé `"nb_ventes"`, de type chaîne de caractères, doit être associée au nombre total d'articles présents dans la liste `ventes`.

Dans un but d'efficacité, il est demandé de ne parcourir qu'une seule fois la liste `ventes`.

Par exemple :

```
>>> ventes = [15,102,64,15,15,15,3,64,7,32,97,64,15]
>>> comptage(ventes)
{15: 5, 102: 1, 64: 3, 3: 1, 7: 1, 32: 1, 97: 1, 'nb_ventes': 13}
>>> comptage([])
{'nb_ventes': 0}
```

Exercice 2 (4 points)

Afin de préparer les négociations annuelles sur la revalorisation salariale, votre entreprise voudrait connaître facilement le nombre de salarié.e.s gagnant moins qu'une somme donnée.

Vous aidez la comptable d'une entreprise à finaliser la fonction `nb_salaires_inferieurs` qui prend en paramètre la liste triée par ordre croissant des salaires octroyés dans l'entreprise, paramètre nommé `salaires` et un entier `val` qui représente une valeur monétaire, fonction qui renvoie l'indice du premier élément de la liste `salaires` dépassant strictement la valeur `val`, si cet indice existe et renvoie la longueur de la liste sinon. Comme la liste `salaires` est supposée triée par ordre croissant, une recherche dichotomique est mise en oeuvre dans la fonction `nb_salaires_inferieurs`.

Ainsi, le renvoi de cette fonction permet de connaître directement et efficacement le nombre d'employé.e.s ayant un salaire strictement inférieur à une valeur donnée.

Voici quelques tests qui vous permettront de vérifier le bon fonctionnement de la fonction finalisée :

```
>>> salaires1 = [1550,1550,1550,1620,1630,1780,1840,1870,1870,1910,2300,2500,2640,2670,2800,3200]
>>> nb_salaires_inferieurs(salaires1,1700)
5
>>> nb_salaires_inferieurs(salaires1,1550)
3
>>> nb_salaires_inferieurs(salaires1,1500)
0
>>> nb_salaires_inferieurs(salaires1,3500)
16
```

Compléter le code ci-dessous puis le tester :

```
def nb_salaires_inferieurs(salaires:list,val:float)->int:
    """salaires est une liste de nombres supposée triée par ordre croissant ; val est un nombre réel.
    Fonction qui recherche par dichotomie l'indice du premier élément de la liste salaires dépassant
    strictement val, si un tel élément existe, et le renvoie.
    Si un tel élément n'existe pas, la fonction renvoielalongueur de la liste salaires."""

    indice_inf = 0 # indice minimal du tableau dans lequel on recherche l'élément
    indice_sup = ... # indice maximal du tableau dans lequel on recherche l'élément
    while indice_inf ... indice_sup:
        indice_mil = ... # attention au fait qu'un indice est forcément un nombre entier !
        if salaires[indice_mil] <= val:
            indice_... = ...
        else:
            if indice_mil==0:
                return ...
            elif ...:
                return indice_mil
            else:
                indice_... = ...
    return ...
# tests :
```