

Fonctions linéaires sur les listes



Le but de cette fiche est de (re)faire quelques exercices types liés aux listes, concernant en particulier les recherches de minimum (ou maximum) avec retour de valeurs ou d'indices, de même que des comptes d'occurrences ou des calculs de moyennes.

Les codes Python demandés peuvent être ceux à produire dans l'exercice 1 de l'épreuve pratique.

Soit `liste1` définie par :

```
liste1 = [42, 76, 21, 72, 71, 71, 58, 47, 70, 42, 42,
          26, 40, 58, 76, 22, 49, 43, 83, 17, 78, 28,
          78, 54, 56, 24, 83, 46, 17, 63, 82, 72, 70,
          29, 41, 49, 78, 28, 52, 34, 42, 71, 74, 68]
```

 Chacune des fonctions définies dans les questions suivantes devront être testées, notamment sur `liste1`, avec des `assert`.

Ensuite, les questions sont indépendantes, et il n'est pas autorisé d'appeler des fonctions définies précédemment pour définir une nouvelle fonction.

D'autre part, les fonctions doivent avoir une complexité linéaire. Autrement dit, il s'agit de parcourir la liste donnée en argument une fois seulement (éventuellement pas entièrement).

Enfin, le but de ces questions est de coder des fonctions sans faire appel à des fonctions ou méthodes de base de Python comme `min`, `index`, `count`, etc. qui sont donc interdites ici.

Si vous pensez à plusieurs méthodes un peu différentes (notamment sur la manière de parcourir les listes), ne pas hésiter à les tester toutes, même si ce n'est pas imposé.

1. Minimum (et maximum)

(a) Quelle est la valeur minimale de `liste1` ?

Expliquer comment l'obtenir.

(b) Définir la fonction `minimum(tableau)` qui prend en argument une liste d'entiers `tableau` et qui retourne la valeur minimale de `tableau`.

(c) Vérifier le résultat obtenu avec la liste `liste2 = [-x for x in liste1]`.

Corriger le code si nécessaire.

(d) Refaire les questions précédentes avec la fonction `maximum(tableau)`.

2. Appartenance

Définir la fonction `appartient(val, tableau)` qui prend en argument une valeur `val` et une liste `tableau` et qui retourne `True` si `val` est un élément de `tableau`, `False` sinon.

3. Nombre d'occurrences

(a) Quel est le nombre d'occurrences (d'apparitions) du nombre 78 dans `liste1` ?

Expliquer comment l'obtenir.

(b) Définir la fonction `nombre_occurrences(val, tableau)` qui prend en argument une valeur entière `val` et une liste d'entiers `tableau` et qui retourne le nombre d'occurrences de `val` dans `tableau`.

4. Liste d'indices d'une valeur

(a) Quelle est la liste des indices des occurrences de 78 dans `liste1` ?

Expliquer comment l'obtenir.

- (b) Définir la fonction `liste_indices(val, tableau)` qui prend en argument une valeur entière `val` et une liste d'entiers `tableau` et qui retourne la liste des indices des valeurs de `tableau` égales à `val`.

5. Liste d'indices du minimum

Définir la fonction `indices_min(tableau)` qui prend en argument une liste d'entiers `tableau` et qui retourne la liste des indices des occurrences de la valeur minimale de `tableau`.

Il est rappelé que la complexité de la fonction doit être linéaire, donc que la liste ne doit être parcourue qu'une seule fois.

6. Indice de la première occurrence d'une valeur

Définir la fonction `premiere_occurrence(val, tableau)` qui prend en argument une valeur entière `val` et une liste d'entiers `tableau` et qui retourne l'indice de la première occurrence de `val` dans `tableau`.

7. Indice de la dernière occurrence d'une valeur

- (a) Définir la fonction `derniere_occurrence(val, tableau)` qui prend en argument une valeur entière `val` et une liste d'entiers `tableau` et qui retourne l'indice de la dernière occurrence de `val` dans `tableau`.
- (b) Réfléchir à une autre manière de faire et donner une seconde définition de la fonction `derniere_occurrence(val, tableau)`.

8. Moyenne

- (a) On admettra que la moyenne des valeurs de `liste1` vaut 53.25.
Expliquer comment l'obtenir.
- (b) Définir la fonction `moyenne(tableau)` qui prend en argument une liste d'entiers `tableau` et qui retourne la moyenne des valeurs de `tableau`.

Ces questions supplémentaires font manipuler, en plus des listes, les dictionnaires.

8. Dictionnaire des occurrences des valeurs

Définir la fonction `dict_occurrences(tableau)` qui prend en argument une liste d'entiers `tableau` et qui retourne un dictionnaire dont les clés sont les valeurs de `tableau` et dont les valeurs sont les occurrences de ces valeurs dans `tableau`.

```
d1 = {42: 4, 76: 2, 21: 1, 72: 2, 71: 3, 58: 2, 17: 2,
      70: 2, 40: 1, 22: 1, 49: 2, 43: 1, 83: 2, 47: 1,
      78: 3, 28: 2, 54: 1, 56: 1, 24: 1, 46: 1, 26: 1,
      63: 1, 82: 1, 29: 1, 41: 1, 52: 1, 34: 1, 74: 1, 68: 1}
assert dict_occurrences(liste1) == d1
```

9. Dictionnaire des indices des valeurs

Définir la fonction `enumere(tableau)` qui prend en argument une liste d'entiers `tableau` et qui retourne un dictionnaire dont les clés sont les valeurs de `tableau` et dont les valeurs sont les listes des indices de ces valeurs.

```
d2 = {42: [0, 9, 10, 40], 76: [1, 14], 21: [2], 72: [3, 31],
      71: [4, 5, 41], 58: [6, 13], 47: [7], 70: [8, 32],
      26: [11], 40: [12], 22: [15], 49: [16, 35], 43: [17],
      83: [18, 26], 17: [19, 28], 78: [20, 22, 36],
      28: [21, 37], 54: [23], 56: [24], 24: [25], 46: [27],
      63: [29], 82: [30], 29: [33], 41: [34], 52: [38],
      34: [39], 74: [42], 68: [43]}
assert enumere(liste1) == d2
```