

Nom : ..... Prénom : .....

**Exercice 1.** Écrire une fonction `occurrences` qui étant donné un tableau `tab` renvoie un dictionnaire structuré de la manière suivante :

- l'ensemble des clés du dictionnaire est l'ensemble des valeurs de `tab` ;
- à chaque clé `k` est associée la liste des indices des occurrences de `k` dans `tab`.

Votre fonction aura une complexité **linéaire**.

Code python

```
1 def occurrences(tab):
2     """ [int] -> {int:[int]}
3     Compte les occurrences des éléments de tab """
4     .....
5     .....
6     .....
7     .....
8     .....
9     .....
10    .....
11    .....
12    .....
```

**Exercice 2. 1.** Écrire une fonction `minimum` qui étant donné un tableau `tab` renvoie un tuple constitué de deux éléments :

- `mini`, qui est le minimum des valeurs du tableau `tab` ;
- `indices`, qui est la **liste** des indices où ce minimum est atteint.

Par exemple :

Code python

```
1 print(minimum([5, 6, 7]))
2 print(minimum([7, 8, 9, 7, 8, 9]))
```

 Résultat

```
(5, [0])
(7, [0, 3])
```

Code python

```
1 def minimum(tab):
2     """ [int] -> (int, [int]) """
3     .....
4     .....
5     .....
6     .....
7     .....
8     .....
9     .....
10    .....
11    .....
12    .....
13    .....
14    .....
```

**2.** Comment modifier le code de la fonction `minimum` afin que la liste `indices` renvoyée soit triée par **ordre décroissant** (une seule ligne à modifier) ?

.....  
.....


**Exercice 3.** Un météorologue a relevé les températures au lever du jour dans sa rue. Il souhaite déterminer la durée de la plus longue période de gelées consécutives durant ces relevés. On souhaite écrire une fonction `gelees` qui étant donné une liste d'entiers `temperatures` renvoie la longueur de la plus longue séquence de nombres négatifs ou nuls consécutifs dans la liste. Si la liste est vide, la fonction renverra la valeur 0. On rappelle que l'eau gèle à partir de 0°C inclus.

1. Indiquer ce que doit renvoyer chacune des intructions ci-dessous.

- a. `gelees([])` renvoie .....
- b. `gelees([5, 2, 1, 3, 2, 1, 1])` renvoie .....
- c. `gelees([5, 3, 4, -1, -2, 0, 1, -4, -5])` renvoie .....

2. Écrire le code d'une fonction `gelees` qui réponde à la question de l'énoncé. Vous écrirez convenablement la documentation de la fonction qui comportera la signature de la fonction ainsi qu'une description de son effet.

**Indication.** Vous parcourrez le tableau `temperatures` et utiliserez deux variables : une variable `duree_gel` qui indiquera la durée de la période de gel (si il y en a une) en cours pour l'élément d'indice `i`, une variable `plus_grande_duree` qui indique la plus grande valeur prise par la variable `duree_gel`. Ainsi, pour le tableau `[4, 3, -1, -2, 2, 3, -4, -5, -6, 2, -1, -1, 2]` on a :

 Résultat

```
Indice : 0, duree_gel=0, plus_grande_duree=0
Indice : 1, duree_gel=0, plus_grande_duree=0
Indice : 2, duree_gel=1, plus_grande_duree=1
Indice : 3, duree_gel=2, plus_grande_duree=2
Indice : 4, duree_gel=0, plus_grande_duree=2
Indice : 5, duree_gel=0, plus_grande_duree=2
Indice : 6, duree_gel=1, plus_grande_duree=2
Indice : 7, duree_gel=2, plus_grande_duree=2
Indice : 8, duree_gel=3, plus_grande_duree=3
Indice : 9, duree_gel=0, plus_grande_duree=3
Indice : 10, duree_gel=1, plus_grande_duree=3
Indice : 11, duree_gel=2, plus_grande_duree=3
Indice : 12, duree_gel=0, plus_grande_duree=3
```

Code python

1	.....
2	.....
3	.....
4	.....
5	.....
6	.....
7	.....
8	.....
9	.....
10	.....
11	.....
12	.....
13	.....
14	.....
15	.....
16	.....

3. Quelle est la complexité de votre fonction `gelees` en fonction de la taille du tableau `temperatures` ? Aucune justification n'est attendue. ....