

Vos fonctions doivent être commentées et documentées. Vos réponses doivent être justifiées.

Exercice 1 : Questions diverses

Les questions suivantes sont indépendantes.

1. Écrire une fonction **mini**(L) qui prend en paramètre une liste d'entiers L et renvoie le minimum de la liste L ainsi qu'un indice de ce minimum.
2. Écrire une fonction **comptage**(L) qui prend en argument une liste L et renvoie un dictionnaire permettant de connaître le nombre d'occurrences de chaque élément de la liste L .

Voici un exemple de ce que l'on doit obtenir :

```
1 >>> L = [1, 4, 4, 6, 1, 0, 7, 4]
2 >>> comptage(L)
3 {1: 2, 4: 3, 6: 1, 0: 1, 7: 1}
```

3. Écrire la fonction **tribulles**(L) qui prend en paramètre une liste d'entiers L et renvoie la liste triée en suivant l'algorithme du tri bulle. Donner sans justification la complexité de cet algorithme.
4. Écrire les lignes de commandes qui permettent de tracer la fonction $x \mapsto \sqrt{\sin(x^2 - x + 1) + 4}$ sur l'intervalle $[-7, 3]$.
5. On considère la fonction suivante :

```
1 def f(n):
2     """ n est un entier naturel """
3     p = 1
4     i = 0
5     while i < n:
6         p = p * 2
7         i = i + 1
8     return(p)
```

- (a) Que calcule cette fonction ?
- (b) Justifier qu'elle se termine en donnant un variant de boucle.
- (c) Justifier qu'elle renvoie le résultat voulu en donnant un invariant de boucle.

Exercice 2 : Tri des crêpes

• Léa, qui maîtrise déjà l'élaboration des cookies, tente aujourd'hui de faire des crêpes. Elle a fait n crêpes (où $n \in \mathbb{N}^*$). Ces crêpes, toutes de diamètres différents sont empilées dans une assiette. Léa aimerait les trier par taille croissante, la plus grande devant être en-dessous du tas. Le seul ustensile à proximité est une spatule que l'on peut glisser entre les crêpes afin de retourner toutes les crêpes au-dessus de la spatule. C'est la seule opération que l'on a le droit d'effectuer lors du tri de la pile de crêpes.



• Les tailles des crêpes sont numérotées de 1 à n . Voici un exemple où l'on parvient à remettre la pile dans le bon ordre après 6 manipulations.

3	5	1	4	2	3	1
2	2	4	1	3	2	2
5	3	3	3	1	1	3
4	4	2	2	4	4	4
1	1	5	5	5	5	5

• La position de la spatule est représentée par le trait noir. Par exemple, pour passer de la pile 1 à la pile 2, on place la spatule sous la crêpe de taille 5 et on retourne tout le bloc de crêpes au-dessus de la spatule.

• On représente les tailles des crêpes par une liste. La liste qui correspond à la première pile de l'exemple ci-dessus est $L = [3, 2, 5, 4, 1]$.

1. Écrire une fonction *grande*(L, i) qui renvoie l'indice de la plus grande crêpe parmi les i premières crêpes de la pile L . Par exemple :

```
1 >>> grande([3, 2, 5, 4, 1], 3)
2 2 # c'est bien l'indice de la plus grande crêpe parmi les trois premières
```

2. Écrire une fonction *retourner*(L, i) qui prend en paramètre une liste L et un entier i et qui réalise le retournement de la pile de crêpes L lorsque la spatule est placée à la position i . La modification de la liste se fera en place, c'est-à-dire que l'on ne formera pas une liste annexe. Par exemple, la première manipulation que l'on effectue ci-dessus correspond à :

```
1 >>> retourner([3, 2, 5, 4, 1], 2)
2 [5, 2, 3, 4, 1]
```

3. (a) Expliciter une méthode permettant de trier la pile de crêpes. Pour comprendre cet algorithme, on pourra s'aider de l'exemple de l'énoncé.
- (b) Programmer la fonction *tricrepes*(*L*) qui réalise le tri de la pile de crêpes selon la méthode présentée à la question précédente.
- (c) Justifier qu'il est toujours possible de trier un tas de n crêpes en $2n - 3$ étapes. Il est possible que votre algorithme fasse quelques étapes de plus si vous n'avez pas optimisé la fin mais ce n'est pas gênant.

Cette borne de $2n - 3$ n'est pas optimale. Bill Gates et Christos Papadimitriou ont démontré que le nombre minimal de retournements requis pour toute pile de n crêpes est compris entre $\frac{15n}{14}$ et $\frac{18n}{11}$. La valeur exacte en fonction de n est toujours un problème ouvert.

4. Comme dans tout algorithme de tri, la seule opération que l'on va dénombrer est la comparaison entre deux éléments de la liste. Donner la complexité de votre algorithme dans le pire des cas.
5. **Variante des crêpes brulées.** Léa a été un peu inattentive car une face de chaque crêpe est cramée. Dans cette variante, les crêpes sont encore placées dans une assiette, la face brulée pouvant être en haut ou en bas de chaque crêpe. On souhaite cette fois encore trier le tas de crêpes à l'aide de la spatule mais en ajoutant la condition que toutes les faces cramées soient en bas (pour cacher le désastre). Vous pourrez :
 - étudier la faisabilité de ce tri
 - choisir vous-même la façon de représenter les données en Python
 - proposer des fonctions réalisant ce tri.