

ALGORITHMES DE BASE SUR LES LISTES

1. Algorithme de recherche dans une liste

Exercice 1. Écrire une fonction répondant à la question : l'objet x appartient-il à la liste L ?
(la place ci-dessous est pour la correction qui sera à connaître !)

1.1 Recherche d'un élément dans une liste non triée

```
def recherche(x, L):  
    """ Répond à la question : l'objet x appartient-il  
        à la liste L ? """
```

Exercice 2.

1. Écrire un invariant de boucle qui décrit la situation à la fin du k ième tour de boucle.
2. Comment modifier cette fonction pour qu'elle retourne l'indice éventuel de première apparition de x ? De dernière apparition de x ?

1.2 Recherche du maximum

Soit L une liste de nombres. Le but est de déterminer son maximum M .

```
def recherche_max(L):  
    """ Renvoie le maximum de la liste de nombres L """  
    n = len(L)  
    M = L[0]  
    for i in range(1, n):  
        if L[i] > M:  
            M = L[i]  
    return M
```

Exercice 3.

1. Est-il important d'initialiser `M` à `L[0]` ?
2. Ajouter des commentaires pertinents dans cet algorithme.
3. Comment déterminer le minimum de `L` ?
4. Modifier cet algorithme pour qu'il affiche l'indice de première apparition du maximum.
5. Modifier cet algorithme pour qu'il affiche l'indice de dernière apparition du maximum.
6. Modifier cet algorithme pour qu'il affiche la liste des indices d'apparition du maximum.

2. Calcul de la moyenne et de la variance

Définition. Soit n en entier. Soit $x_0, \dots, x_{n-1}$ une liste contenant n nombres.

La **moyenne** de cette liste est

$$m = \frac{x_0 + \dots + x_{n-1}}{n} = \frac{1}{n} \sum_{i=0}^{n-1} x_i$$

La **variance** de cette liste de nombres est

$$V = \frac{1}{n} \sum_{i=0}^{n-1} (x_i - m)^2 \quad \text{qui est aussi égal à} \quad \left(\frac{1}{n} \sum_{i=0}^{n-1} x_i^2 \right) - m^2$$

où m est la moyenne précédente.

Exercice 4. **Algorithmes à connaître**

1. Écrire en **Python** une fonction **somme** de paramètre une liste de nombres `L` et qui renvoie la somme des ses éléments.
2. En déduire une fonction **moyenne** de paramètre une liste de nombres `L` et qui renvoie la moyenne des ses éléments.

Dès lors, en utilisant les fonctions **somme** et **moyenne** vues dans l'exercice précédent, on propose la fonction **variance** suivante :

```
def variance(L):  
    """ Renvoie la variance de la liste de nombres L """  
    n = len(L)  
    m = moyenne(L) # calcul de la moyenne  
    V = [(x-m)**2 for x in L]  
        # liste à sommer pour la variance  
    return somme(V)/n
```