

# Fiche cours 8 : tris de listes

---

## Points abordés

- Listes.
  - Tri de listes.
- 

## 1 Introduction

Dans de nombreuses structures informatiques (ou non), il est important que les données soient rangées (organisées) d'une certaine façon. Par exemples, dans un agenda les dates sont triées dans l'ordre chronologique et dans un dictionnaire les mots sont triés dans l'ordre lexicographique. Sans une manière d'organiser les données, certaines opérations comme la recherche d'un élément peuvent s'avérer plus complexes : si tous les mots du dictionnaire étaient écrits dans un ordre aléatoire, trouver un mot donné pourrait être compliqué.

Les listes sont une des structures fondamentales en informatique : elles permettent de stocker et de modifier des données. En supposant qu'il existe une relation d'ordre sur ces données, il pourrait être intéressant de pouvoir les trier. C'est l'objet des différents algorithmes présentés dans la suite.

## 2 Algorithmes étudiés

Nous considérerons deux algorithmes de tri. Ils sont considérés comme "naïfs", car leur efficacité n'est pas optimale.

### 2.1 Le tri par insertion

Supposons que l'on dispose d'une liste triée  $L$ . Si on sait insérer un élément  $x$  dans  $L$  tout en gardant la liste triée, alors on sait trier une liste de proche en proche. En effet, une liste à un élément est triée. Il suffit alors d'insérer successivement les éléments de  $L$  pour trier la liste.

**Exemple 1.** Pour  $L=[3, 1, 3, 2, 9, 8, 7]$ , voici la construction de la liste triée :

- $M=[]$  on insère 3 ;
- $M=[3]$  on insère 1 ;
- $M=[1,3]$  on insère 3 ;
- $M=[1,3,3]$  on insère 2 ;
- $M=[1,2,3,3]$  on insère 9 ;
- $M=[1,2,3,3,9]$  on insère 8 ;
- $M=[1,2,3,3,8,9]$  on insère 7 ;
- $M=[1,2,3,3,7,8,9]$  on a fini.

- Pour insérer dans une liste triée, on procède de la manière suivante :
- on parcourt la liste jusqu'à trouver la position de l'élément à insérer,
  - on insère l'élément.

## 2.2 Le tri à bulle

Le tri à bulle consiste à "remonter" les petits éléments en effectuant des échanges.

- On commence par la fin de la liste ;
- on compare le dernier élément au précédent ;
- s'il est plus petit, on les échange.
- Sinon, on considère l'avant dernier élément que l'on compare avec le précédent, et ainsi de suite.

À l'issue de la première remontée, le plus petit élément de la liste est bien positionnée.

**Exemple 2.** Appliquons le tri à bulle à la liste  $L=[3,1,3,2,9,8,7]$ , en indiquant les échanges successives.

- $L=[3,1,3,2,9,8,7]$  comme  $7 < 8$ , on les échange ;
- $L=[3,1,3,2,9,7,8]$  comme  $7 < 9$ , on les échange ;
- $L=[3,1,3,2,7,9,8]$  comme  $7 > 2$ , on ne fait rien ;
- $L=[3,1,3,2,7,9,8]$  comme  $2 < 3$ , on les échange ;
- $L=[3,1,2,3,7,9,8]$  comme  $2 > 1$ , on ne fait rien ;
- $L=[3,1,2,3,7,9,8]$  comme  $1 < 3$ , on les échange ;
- $L=[1,3,2,3,7,9,8]$ .

On a effectué ici la première remontée. On constate bien que le premier élément est bien le plus petit. Pour trier la liste  $L$ , on applique cet algorithme de remontée 7 fois.