



**Taki Academy**  
[www.takiacademy.com](http://www.takiacademy.com)

# Algo & prog

Classe : 4<sup>ème</sup> sciences de l'informatique

Résumé

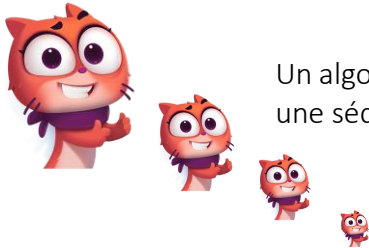
Chapitre : Les algorithmes de tri

📍 Sousse (Khezama - Sahloul) Nabeul / Sfax / Bardo / Menzah El Aouina /  
Ezzahra / CUN / Bizerte / Gafsa / Kairouan / Medenine / Kébili / Monastir /  
Gabes / Djerba



# Les algorithmes de tri

## 1- Définition :



Un algorithme de tri est une suite d'instructions servant à **ordonner** (ranger) une séquence d'éléments de façon **croissante** ou **décroissante**

### Exemples :

- Séquence triée : 12 – 15 – 88 – 121 – 122 – 714 – 901 – 2510
- Séquence non triée : 15.5 – 19 – 10 - 201 - 3 – 155

## REMARQUE

Nombreux algorithmes de tri existent voici quelques exemples :  
Tri à bulles, Tri par sélection, Tri par insertion, Tri Shell, Tri rapide, Tri à peigne, Tri fusion, Tri comptage, Tri Gnome, Tri stupide, ...

## 2- Tri à bulles :

### 2.1/ Principe (tri croissant d'un tableau T de N cases) :

Cette méthode consiste à faire remonter le plus grand élément du tableau en comparant les éléments successifs, elle fonctionne comme suit :

- 1- On compare le premier **pair d'éléments T[0] et T[1]**.
- 2- Si  $T[0] > T[1]$  alors on permute T[0] et T[1], aller au pair suivant et répéter les étapes 1 et 2 jusqu'à comparer le dernier pair T[N-2] et T[N-1]. A la fin de ce premier parcours on aura passé le plus grand élément du tableau vers sa place finale qui est le (N-1)<sup>ème</sup> case du tableau.
- 3- On recommence cette opération en parcourant de 0 à N-2 puis de 0 à N-3 et ainsi de suite.
- 4- On arrête le traitement si on arrive au dernier élément du tableau ou le tableau devient trié.

$TAB = \text{Tableau de } N \text{ entier}$

### 2.2/ Algorithme de tri à bulles :

Procédure Tri\_bulles (@ T : TAB , N : entier)

DEBUT

**Répéter**

**test** ← Faux

Pour i de 0 à N-2 faire

Si  $T[i] > T[i+1]$  alors

aux ← T[i]

T[i] ← T[i+1]

T[i+1] ← aux

**test** ← Vari

Finsi

Fin pour

N ← N-1

**Jusqu'à** (N = 1) ou (**test** = Faux)

FIN

TDOL :

Objet	Type / Nature
aux, i test	Entier Booléen

### 2.3/ Autre version de tri à bulles :

Procédure Tri\_bulles (@T: TAB , N : entier)

DEBUT

Pour i de 0 à N-1 Faire

Pour j de 0 à N-i-2 Faire

Si  $T[j] > T[j+1]$  Alors

aux ← T[j]

T[j] ← T[j+1]

T[j+1] ← aux

Fin Si

Fin Pour

Fin Pour

FIN

TDOL

Objet	Type / nature
i, j, aux	Entier

### 3- Tri par sélection :

#### 3.1/ Principe (tri croissant d'un tableau T de N cases) :

Cette méthode consiste à :

- 1- Trouver l'indice ou la position (pm) du plus petit élément du tableau.
- 2- **Placer le plus petit élément (T[pm]) à sa position finale** (la première position)
- 3- Rechercher l'indice du second plus petit élément
- 4- Le placer à sa position finale (deuxième position)
- 5- Répéter le traitement précédent (3 et 4) jusqu'à ce que le tableau soit trié.

#### 3.2/ Algorithme de tri par sélection :

*TAB = Tableau de N entier*

Procédure Tri\_selection (@T: TAB , N : entier)

DEBUT

Pour i de 0 à N-2 faire

#on commence par chercher la position du minimum

pm ← i

Pour j de i+1 à N-1 faire

Si (T[j] < T[pm]) alors

pm ← j

Finsi

Fin pour

Si (pm ≠ i) Alors

aux ← T[i]

T[i] ← T[pm]

T[pm] ← aux

Fin si

Fin Pour

FIN

TDOL

Objet	Type / nature
i,j,pm, aux	entier

## 4- Tri par insertion :

### 4.1/ Principe (tri croissant d'un tableau T de N cases) :

- 1- On commence par le deuxième élément du tableau c'est-à-dire T[1].
- 2- On compare l'élément choisi (v) avec tous ses précédents dans la liste (la partie gauche du tableau) afin de l'insérer dans la bonne position après avoir décalé tous les éléments qui sont supérieurs à (v) vers la droite.
- 3- Répéter l'étape 2 pour l'élément suivant jusqu'à arriver au dernier élément du tableau.

### 4.2/ Algorithme de tri par insertion :

*TAB = Tableau de N entier*

Procédure Tri\_insertion (@T : TAB, N : entier)

DEBUT

Pour i de 1 à N-1 faire

$v \leftarrow T[i]$

$j \leftarrow i$

Tant que (T[j-1]>v) et (j>0) faire

$T[j] \leftarrow T[j-1]$

$j \leftarrow j-1$

Fin tant que

$T[j] \leftarrow v$

Fin pour

FIN

TDOL

Objet	Type / Nature
i,j,v	Entier

## 5- Tri shell :

### 5.1/ Présentation :

Le tri par shell est une amélioration de tri par insertion proposée par **Donald L.Shell** en 1959. Le tri par shell se base sur le calcul de **pas** (gap).

**Question** : Mais qu'est-ce que le pas et comment le calculer ?

**Réponse** : Donald Shell a proposé la formule (suite) suivante pour calculer le pas : (avec N est la taille du tableau)

$$\begin{cases} U_0 = 1 \\ U_{n+1} = 3 * U_n + 1 \end{cases}$$

## 5.2/ Exemple de calcul de pas pour un tableau de 100 éléments :

Pour calculer le pas d'un tableau de taille  $N=100$  on procède comme suit :

**Etape1 : déterminer la valeur maximale du pas**

$$U_0=1$$

$$U_1=3*U_0 + 1 = 3*1 + 1 = 4$$

$$U_2=3*U_1 + 1 = 3*4 + 1 = 13$$

$$U_3=3*U_2 + 1 = 3*13 + 1 = 40$$

$$U_4=3*U_3 + 1 = 3*40 + 1 = 121 \text{ (stop car on dépasse la taille du tableau)}$$

**Etape2 : Les valeurs du pas pour trier le tableau sont :**

$$\text{Pas1}=121 \text{ div } 3 = 40$$

$$\text{Pas2}=40 \text{ div } 3 = 13$$

$$\text{Pas3}=13 \text{ div } 3 = 4$$

$$\text{Pas4}=4 \text{ div } 3 = 1 \text{ (stop car on ne peut pas faire un pas moins que 1)}$$

*TAB = Tableau de N entier*

## 5.3/ Algorithme de tri Shell :

Procédure Tri\_Shell (@ T :TAB, N :entier)

Début

$p \leftarrow 0$

**Tant que** ( $p < N$ ) **faire**

$p \leftarrow (3*p+1)$

**Fin tant que**

Calcul de pas maximal

**Tant que** ( $p \neq 0$ ) **faire**

$p \leftarrow p \text{ div } 3$

**Pour** i de p à N-1 **faire**

$v \leftarrow T[i]$

$j \leftarrow i$

**Tant que** ( $j > p-1$ ) et ( $T[j-p] > v$ ) **faire**

$T[j] \leftarrow T[j-p]$

$j \leftarrow j-p$

**Fin tant que**

$T[j] \leftarrow v$

**Fin pour**

**Fin tant que**

**Fin**

TDOL

Objet	Type / Nature
p,i,j,v	entier

## Remarque

Le tri par insertion est un cas particulier du tri Shell où le pas = 1







**Taki Academy**  
[www.takiacademy.com](http://www.takiacademy.com)



Sousse (Khezama - Sahloul) Nabeul / Sfax / Bardo / Menzah El Aouina /  
Ezzahra / CUN / Bizerte / Gafsa / Kairouan / Medenine / Kébili / Monastir /  
Gabes / Djerba



[www.takiacademy.com](http://www.takiacademy.com)



73.832.000