

Le but de ces algorithmes est de permettre de trier des éléments le plus rapidement possible. Nous allons utiliser ces algorithmes dans un contexte de tableau à 1 dimension :

0	1	2	3	4
5	3	1	2	4

## 1) Le tri par sélection

👉 Voici le principe (pour un tri croissant) :

Chercher l'indice de l'élément le plus petit : 2

Permuter le contenu de la case 2 avec le contenu de la case 0

0	1	2	3	4
1	3	5	2	4
partie triée		partie non triée		
↔		↔		

Chercher l'indice de l'élément le plus petit dans la partie non triée : 3

Permuter le contenu de la case 3 avec le contenu de la case 1

0	1	2	3	4
1	2	5	3	4
partie triée		partie non triée		
↔		↔		

Chercher l'indice de l'élément le plus petit dans la partie non triée : 3

Permuter le contenu de la case 3 avec le contenu de la case 2

0	1	2	3	4
1	2	3	5	4
partie triée			partie non triée	
↔			↔	

Chercher l'indice de l'élément le plus petit dans la partie non triée : 4

Permuter le contenu de la case 4 avec le contenu de la case 3

0	1	2	3	4
1	2	3	4	5

Le tableau est maintenant trié par ordre croissant !

👉 Travail à faire :

- Etablir la fonction qui permet d'effectuer le tri par sélection d'un tableau passé en paramètre (en entrée le tableau non trié, en sortie, le tableau trié).

## 2) Le tri par permutation

0	1	2	3	4
5	3	1	2	4

🔧 Voici le principe (pour un tri croissant)

On se positionne sur la case à l'indice 1 (donc la 2<sup>ème</sup> case)

Si le contenu de la case 1 est plus petit que le contenu de la case 0 Alors

On les permute

Sinon On arrête

0	1	2	3	4
3	5	1	2	4

On se positionne sur la case à l'indice 2

Si le contenu de la case 2 est plus petit que le contenu de la case 1 Alors

On les permute

On descend d'une case et on re-teste (case 1 avec case 0)

Sinon On arrête

0	1	2	3	4
3	1	5	2	4

0	1	2	3	4
1	3	5	2	4

On se positionne sur la case à l'indice 3

Si le contenu de la case 3 est plus petit que le contenu de la case 2 Alors

On les permute

On descend d'une case et on re-teste (2 et 1 ; 1 et 0)

Sinon On arrête

0	1	2	3	4
1	3	2	5	4

0	1	2	3	4
1	2	3	5	4

On se positionne sur la case à l'indice 4

Si le contenu de la case 4 est plus petit que le contenu de la case 3 Alors

On les permute

On descend d'une case et on re-teste (3 et 2 ; 2 et 1 ; 1 et 0)

Sinon On arrête

0	1	2	3	4
1	2	3	4	5

🔧 Travail à faire :

- Etablir la fonction qui permet d'effectuer le tri par permutation d'un tableau passé en paramètre.

### 3) Le tri à bulles

	0	1	2	3	4	5
tab	2	6	1	4	5	3

👉 Voici le principe (pour un tri croissant)

1 : se positionner sur la case à l'indice 0

a : Comparer le contenu avec celui de la case à l'indice 1 :

si le contenu de la deuxième case est plus petit, permuter les valeurs

tab	2	6	1	4	5	3
-----	---	---	---	---	---	---

b : Comparer le contenu avec celui de la case à l'indice 2 :

si le contenu de la troisième case est plus petit, permuter les valeurs

tab	1	6	2	4	5	3
-----	---	---	---	---	---	---

c : Faire de même avec toutes les cases jusqu'à la dernière...

👉 **A ce stade la première case du tableau contient maintenant la plus petite valeur !**

2 : Se positionner sur la case à l'indice 1 :

a : Comparer le contenu avec la case à l'indice 2 :

si le contenu de la troisième case est plus petit, permuter les valeurs

tab	1	2	6	4	5	3
-----	---	---	---	---	---	---

b : Faire de même avec toutes les cases jusqu'à la dernière...

👉 **A ce stade la deuxième case du tableau contient maintenant la 2<sup>ème</sup> valeur la plus petite !**

3 : Se positionner sur la case à l'indice 2...

👉 Travail à faire :

- Etablir la fonction qui permet d'effectuer un tri à bulles d'un tableau passé en paramètre.