

# Algorithmes de tri

**Sébastien Jean**

IUT de Valence  
Département Informatique

v1.0, 16 décembre 2025

- **Trier** = **arranger les éléments** d'une collection selon une **relation d'ordre total**
- Trier permet de pouvoir ensuite :
  - **Traiter les éléments dans l'ordre** voulu
  - **Faciliter la recherche** d'un élément

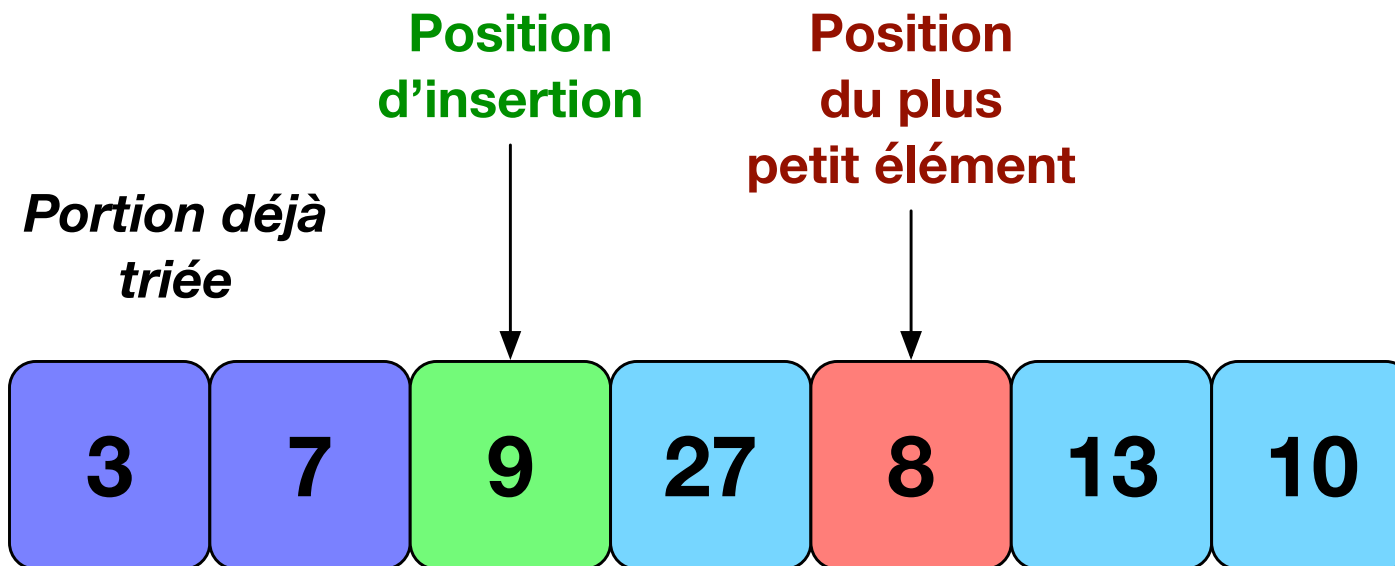


# Algorithmes de tri

- Quelques algorithmes de tris usuels
  - Tri par sélection
  - Tri par insertion
  - Tri à bulles
  - Tri rapide
  - Tri fusion
  - Tri par tas
- Complexité en temps et en mémoire (meilleur/pire des cas, en moyenne) différentes
- Possibilité ou non de trier en place, ...

# Tri par sélection : idée générale

- On traite chaque position de gauche à droite, on cherche la position du plus petit élément entre la position (incluse) et la fin du tableau, on permute l'élément à cette position avec celui occupant la première position
- On recommence pour trouver l'élément à stocker à la seconde position, ainsi de suite ...
- ...



# Tri par sélection : algorithme

```
FONCTION tri_selection(t : Tableau de T, n : entier)
    : Tableau de T

    VARIABLE r      : Tableau de T [n]
    VARIABLE i      : entier
    VARIABLE j      : entier
    VARIABLE min    : entier

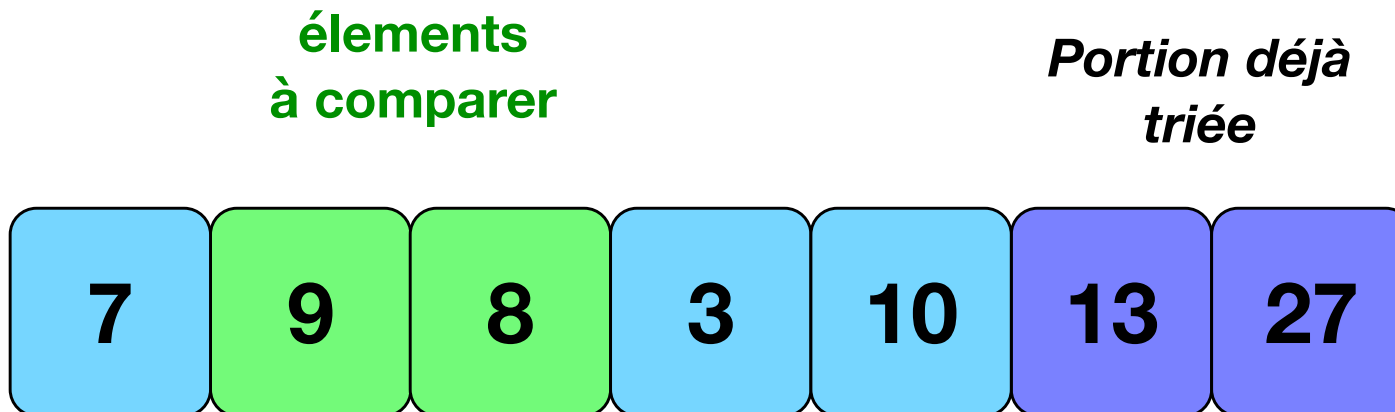
    r ← t
    POUR i de 1 à n-1 PAR PAS DE 1
        min ← i
        POUR j de i+1 à n PAR PAS DE 1
            SI r[j] < r[min] ALORS
                min ← j
            FIN SI
        FIN POUR
        r ← permuter(r, i, min)
    FIN POUR
    RETOURNER r
FIN FONCTION
```

# Tri par sélection : en action



# Tri à bulles : idée générale

- On traite chaque position de gauche à droite, on compare chaque élément avec son voisin de droite, s'il est plus grand on les permute.
- Quand on a parcouru tous les éléments, le plus grand est à la dernière position
- On recommence pour placer l'élément le plus grand du reste à l'avant dernière position, ainsi de suite ...



# Tri bulle

```
FONCTION tri_à_bulles(t : Tableau de T, n : entier)
    : Tableau de T
    VARIABLE r : Tableau de T [n]
    VARIABLE i : entier
    VARIABLE j : entier

    r ← t
    POUR i DE n A 2 PAR PAS DE -1
        POUR j DE 1 i-1 PAR PAS DE 1
            SI r[j] > r[j+1] ALORS
                r ← permuter(r, j, j+1)
            FIN SI
        FIN POUR
    FIN POUR
    RETOURNER r

FIN FONCTION
```

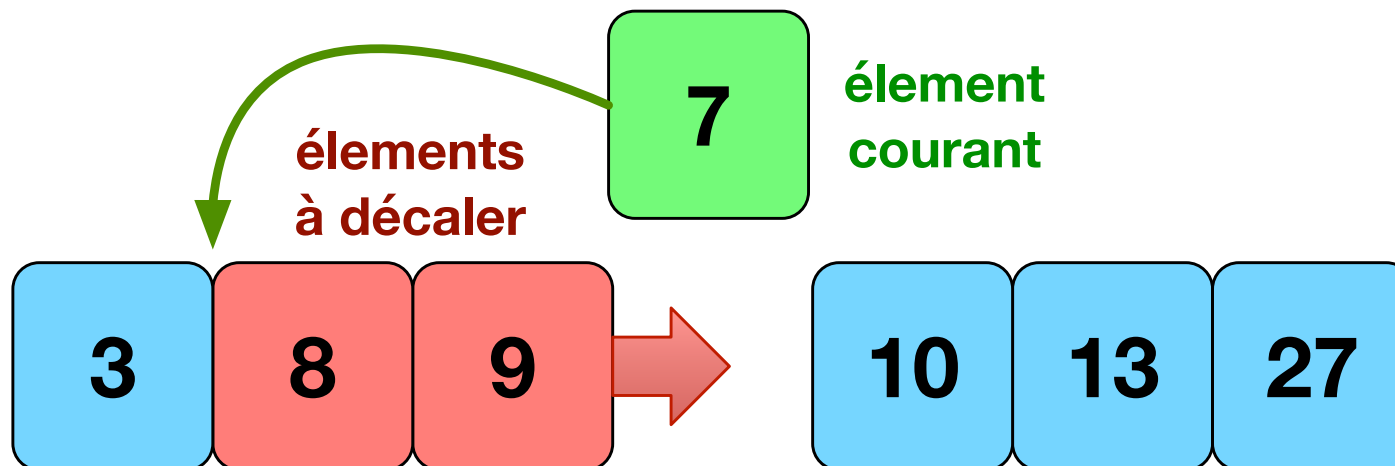


# Tri à bulles : en action



# Tri insertion : idée générale

- On traite chaque position de gauche à droite, on sort l'élément de sa place, on décale vers la droite tous les éléments à sa gauche qui sont plus grands, on le replace dans le trou et ainsi tous les éléments jusqu'à lui (inclus) forment un tableau trié.
- Lorsque la position la plus à droite a été traitée, l'intégralité du tableau est trié
- ...



# Tri insertion

```
FONCTION tri_insertion(t : Tableau de T, n : entier)
                                : Tableau de T
    VARIABLE r                : Tableau de T [n]
    VARIABLE i                : entier
    VARIABLE j                : entier
    VARIABLE tampon           : T

    r ← t
    POUR i DE 2 A n PAR PAS DE 1
        tampon ← r[i]
        j ← i
        TANT QUE j > 1 ET r[j-1] > tampon
            r[j] ← r[j-1]
            j ← j - 1
        FIN TANT QUE
        r[j] ← tampon
    FIN POUR
    RETOURNER r
FIN FONCTION
```

# Tri insertion : en action



# Complexité moyenne en temps

- Tri sélection :  $\mathcal{O}(n^2)$
- Tri à bulles :  $\mathcal{O}(n^2)$
- Tri insertion :  $\mathcal{O}(n^2)$

# Fin !

