

# Trier

G. Aldon - J. Germoni - J.-M. Mény

IREM de Lyon

Mars 2012

Deux tris dans le programme ISN :

- tri par sélection,
- tri par fusion.

# Tri par sélection

Le principe du tri par sélection d'une liste  $T = (T[1], T[2], \dots, T[n])$  :

Pour chaque entier  $j$  ( $1 \leq j \leq n - 1$ ) :

- parcourir les éléments  $T[j], T[j + 1], \dots, T[n]$ , retenir l'indice  $k$  du plus petit.
- placer au rang  $j$  le plus petit des éléments  $T[j], T[j + 1], \dots, T[n]$  (en échangeant  $T[j]$  et  $T[k]$ ).

# Tri par sélection : illustration

2	1	5	0	9	4
---	---	---	---	---	---

# Tri par sélection : illustration

2	1	5	0	9	4
0	1	5	2	9	4

# Tri par sélection : illustration

2	1	5	0	9	4
0	1	5	2	9	4

# Tri par sélection : illustration

2	1	5	0	9	4
0	1	5	2	9	4
0	1	5	2	9	4

## Tri par sélection : illustration

2	1	5	0	9	4
0	1	5	2	9	4
0	1	5	2	9	4
0	1	2	5	9	4

# Tri par sélection : illustration

2	1	5	0	9	4
0	1	5	2	9	4
0	1	5	2	9	4
0	1	2	5	9	4
0	1	2	4	9	5

# Tri par sélection : illustration

2	1	5	0	9	4
0	1	5	2	9	4
0	1	5	2	9	4
0	1	2	5	9	4
0	1	2	4	9	5
0	1	2	4	5	9

# Tri par sélection – Algorithme

**Exercice.** Programmer le tri par sélection.

# Tri par sélection – Algorithme

**Exercice.** Programmer le tri par sélection.

**Entrée :**  $T$  liste de  $n$  nombres.

**Sortie :** liste  $T$  triée

**Traitement :**

Pour  $j$  de 1 à  $n - 1$

  indiceMin :=  $j$

    Pour  $k$  de  $j + 1$  à  $n$

      si  $T[k] < T[j]$  alors indiceMin :=  $k$  finSi

    finPour

  Echange de  $T[j]$  et  $T[\text{indiceMin}]$  si  $j \neq \text{indiceMin}$

  finPour

# Tri par sélection - programme xcas



## Xcas

```
selection(T,deb) :={  
  local k, tmp,j;  
  j :=deb;  
  pour k de deb+1 jusque dim(T)-1 faire  
    si T[k]<T[j] alors j :=k;fsi;  
  fpour;  
  si j!=deb alors tmp :=T[deb];T[deb]=<T[j];T[j]=<tmp;fsi;};;  
triSelection(T) :={  
  local j;  
  pour j de 0 jusque dim(T)-2 faire  
    selection(T,j);  
  fpour;};;
```

# Tri sélection – programme python

## Python

```
def selection(T,debut) :
    indiceDuMin=debut
    for k in range(debut+1,len(T)) :
        if T[k]<T[indiceDuMin] :
            indiceDuMin=k
    if indiceDuMin !=debut :
        T[debut],T[indiceDuMin]=T[indiceDuMin],T[debut]
def triSelection(T) :
    for j in range(0,len(T)-1) :
        selection(T,j)
```

## Complexité : tri par sélection

**Exercice.** Évaluer de façon expérimentale (temps ou nombre d'opérations par compteurs) la complexité du tri par insertion.

## Complexité : tri par sélection

**Exercice.** Évaluer de façon expérimentale (temps ou nombre d'opérations par compteurs) la complexité du tri par insertion.

Complexité expérimentale : fichier SAGE ou fichier XCAS...

## Complexité : tri par sélection

**Exercice.** Évaluer de façon expérimentale (temps ou nombre d'opérations par compteurs) la complexité du tri par insertion.

Complexité expérimentale : second degré

## Complexité : tri par sélection

**Exercice.** Évaluer de façon expérimentale (temps ou nombre d'opérations par compteurs) la complexité du tri par insertion.

Complexité expérimentale : second degré

Nombre de comparaisons :

$$\sum_{j=1}^{n-1} \left( \sum_{k=j+1}^n 1 \right) = \sum_{j=1}^{n-1} (n-j) = \frac{1}{2}n(n-1)$$

Nombre d'échanges : au plus le nombre de comparaisons.