

# Tri par sélection



Algorithme



## Quel est le principe du tri par sélection ?

1. Sélectionner le plus **petit** (ou plus **grand**) élément du tableau.
2. Le ranger dans sa **position adéquate**
3. **Avancer** d'un élément
4. **Répéter** les trois étapes précédentes **jusqu'à ce que le tableau soit trié**

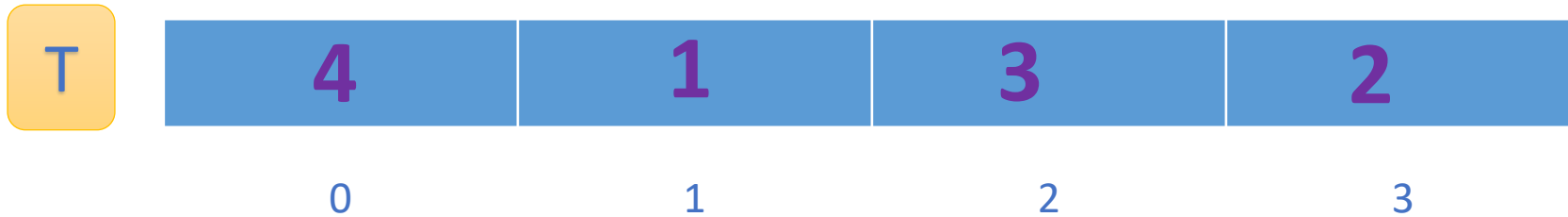


## Exemple ?

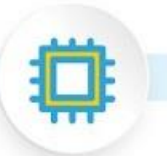




## Trier par sélection



*Ordre croissant*



## Trier par sélection

AUX



T

4

1

3

2

0

1

2

3

*Ordre croissant*



## Trier par sélection

AUX

4

T

1

3

2

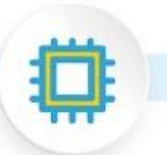
0

1

2

3

*Ordre croissant*



## Trier par sélection

AUX

T

1

4

3

2

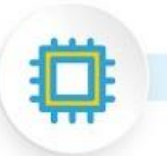
0

1

2

3

*Ordre croissant*



## Trier par sélection

AUX

4

T

1

2

3

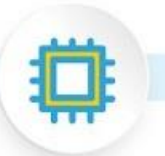
0

1

2

3

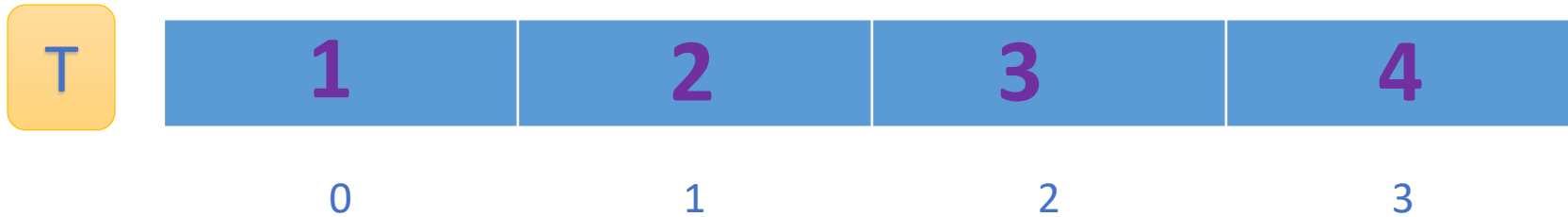
*Ordre croissant*



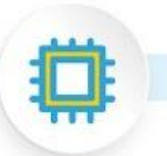


## Trier par sélection

AUX



*Ordre croissant*



# Algorithme

T

15	17	10	25	7	77	3	45
0	1	2	3	4	5	6	7

Procédure tri\_selection (@ t:tab; n:entier)

Début

Pour i de 0 à n-2 faire

pMin ← i

Pour j de i+1 à n-1 faire

Si (t[j] < t[pMin]) alors

alors PMin ← j

fin si

Fin pour



**Ordre croissant**



# Algorithme

T

15	17	10	25	7	77	3	45
0	1	2	3	4	5	6	7

Procédure tri\_selection (@ t:tab; n:entier)

Début

Pour i de 0 à n-2 faire

pMin ← i

Pour j de i+1 à n-1 faire

Si (t[j] < t[pMin]) alors

alors PMin ← j

fin si

Fin pour



**Ordre croissant**



# Algorithme

T	15	17	10	25	7	77	3	45
	0	1	2	3	4	5	6	7

Procédure tri\_selection (@ t:tab; n:entier)

Début

Pour i de 0 à n-2 faire

pMin ← i

Pour j de i+1 à n-1 faire

Si (t[j] < t[pMin]) alors

alors pMin ← j

fin si

Fin pour

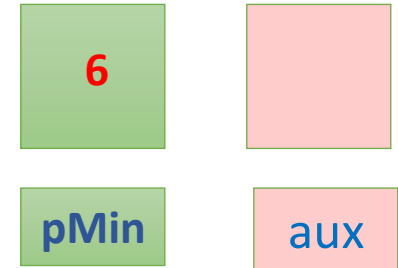
Si (i ≠ pMin) alors

aux ← t[i]

t[i] ← t[pMin]

t[pMin] ← aux

Fin si



Ordre croissant



# Algorithme

T	3	17	10	25	7	77		45
	0	1	2	3	4	5	6	7

Procédure tri\_selection (@ t:tab; n:entier)

Début

Pour i de 0 à n-2 faire

pMin ← i

Pour j de i+1 à n-1 faire

Si (t[j] < t[pMin]) alors

alors pMin ← j

fin si

Fin pour

Si (i ≠ pMin) alors

aux ← t[i]

t[i] ← t[pMin]

t[pMin] ← aux

Fin si

Fin pour

Fin

6	15
pMin	aux

Ordre croissant



# Algorithme

T	3	17	10	25	7	77	15	45
	0	1	2	3	4	5	6	7

Procédure tri\_selection (@ t:tab; n:entier)

Début

Pour i de 0 à n-2 faire

pMin ← i

Pour j de i+1 à n-1 faire

Si (t[j] < t[pMin]) alors

alors pMin ← j

fin si

Fin pour

Si (i ≠ pMin) alors

aux ← t[i]

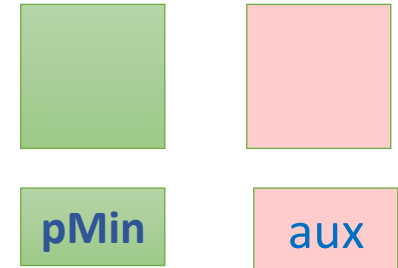
t[i] ← t[pMin]

t[pMin] ← aux

Fin si

Fin pour

Fin



Ordre croissant



# Algorithme

T	3		10	25	7	77	15	45
	0	1	2	3	4	5	6	7

Procédure tri\_selection (@ t:tab; n:entier)

Début

Pour i de 0 à n-2 faire

pMin ← i

Pour j de i+1 à n-1 faire

Si (t[j] < t[pMin]) alors

alors PMin ← j

fin si

Fin pour

Si (i ≠ pMin) alors

aux ← t[i]

t[i] ← t[pMin]

t[pMin] ← aux

Fin si

Fin pour

Fin

4

17

pMin

aux

Ordre croissant



# Algorithme

T	3	7	10	25	17	77	15	45
	0	1	2	3	4	5	6	7

Procédure tri\_selection (@ t:tab; n:entier)

Début

Pour i de 0 à n-2 faire

pMin ← i

Pour j de i+1 à n-1 faire

Si (t[j] < t[pMin]) alors

alors pMin ← j

fin si

Fin pour

Si (i ≠ pMin) alors

aux ← t[i]

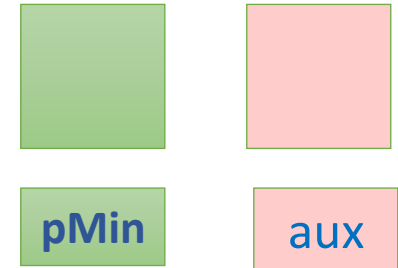
t[i] ← t[pMin]

t[pMin] ← aux

Fin si

Fin pour

Fin



Ordre croissant





# Algorithme

T	5	7	10	20	25	30	45	50
	0	1	2	3	4	5	6	7

Procédure tri\_selection (@ t:tab; n:entier)

Début

Pour i de 0 à n-2 faire

$pMin \leftarrow i$

Pour j de i+1 à n-1 faire

Si  $(t[j] < t[pMin])$  alors  
alors  $pMin \leftarrow j$

fin si

Fin pour

Si  $(i \neq pMin)$  alors

$aux \leftarrow t[i]$

$t[i] \leftarrow t[pMin]$

$t[pMin] \leftarrow aux$

Fin si

Fin pour

Fin

TDOL	
Objet	Type/Nature
i,aux,pMin,j	entier

*Ordre croissant*



*Au revoir*

