



5IPRO

Principes algorithmiques et programmation

Benjamin Delbar



Cours 10

Récap sur les tris

Corrections exercices du cours 8

Exercices récapitulatifs



Récap

Les 3 tris vu?

Quel est leurs similarités?

Que représente le “O”



Les tableaux : tri

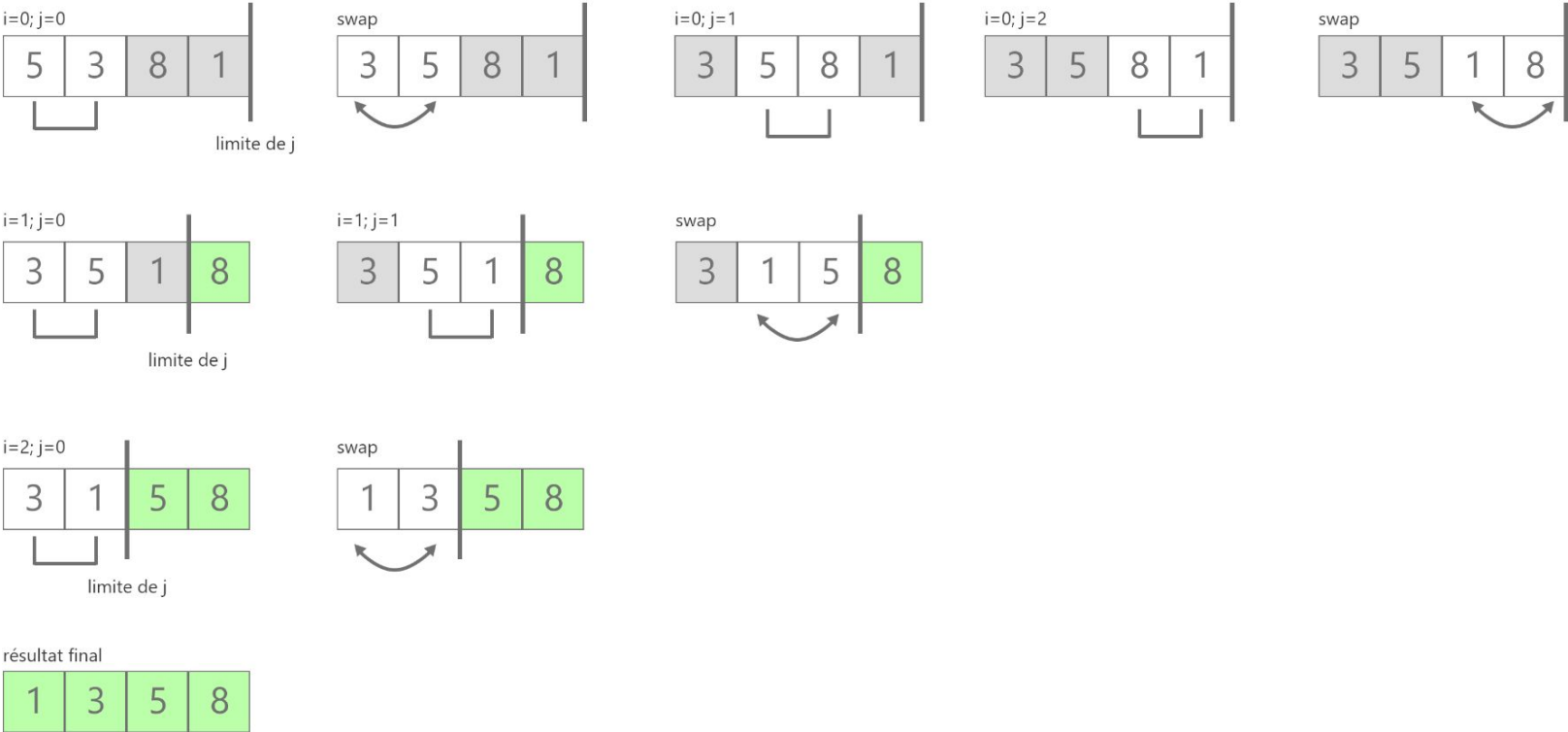
Tri à bulle

$O(n^2)$

Comparer répétitivement l'élément N et $N+1$ et les inverser s'ils ne sont pas ordonnés

- Parcourir le tableau (i)
- Parcourir le tableau de 0 jusqu'à la position $i-1$ (j)
- Si $j > j+1 \Rightarrow$ on échange les 2

Tri à bulle (bubble sort)





Les tableaux : tri

Tri à bulle

Etape 1 : parcourir le tableau

```
myArray = [5, 3, 8, 1];
```

```
for (let i=0; i<myArray.length; i++ ) {  
    //Parcourir le tableau de 0 jusqu'à la position i-1 (j)  
}
```



Les tableaux : tri

Tri à bulle

Etape 2 : parcourir le tableau une seconde fois

```
myArray = [5, 3, 8, 1];

for (let i=0; i<myArray.length; i++ ) {
    for (let j=0; j<myArray.length-i-1; j++) {
        //Si j > j+1 => on échange les 2
    }
}
```



Les tableaux : tri

Tri à bulle

Étape 3 : voir si $j > j+1$

```
myArray = [5, 3, 8, 1];
```

```
for (let i=0; i<myArray.length; i++ ) {  
    for (let j=0; j<myArray.length-i-1; j++) {  
        if (myArray[j] > myArray[j+1]) {  
            //intervertir j et j+1  
        }  
    }  
}
```




Les tableaux : tri

Tri à bulle

Etape 4 : intervertir

```
myArray = [5, 3, 8, 1];

for (let i=0; i<myArray.length; i++ ) {
  for (let j=0; j<myArray.length-i-1; j++) {
    if (myArray[j] > myArray[j+1]) {
      let temp = myArray[j];
      myArray[j] = myArray[j+1]
      myArray[j+1] = temp;
    }
  }
}
```



Les tableaux : tri

Tri à bulle

codepen

<https://codepen.io/Snaj/pen/MWEPPER>



Tri par insertion



Les tableaux : tri

Tri par insertion

$O(n^2)$

On définit une partie gauche du tableau comme triée (sorted),

On parcourt les éléments non triés (unsorted) et on les insère à la bonne position dans la partie triée



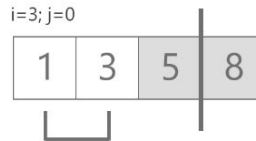
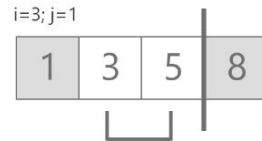
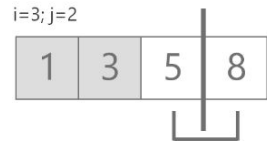
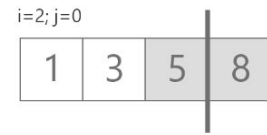
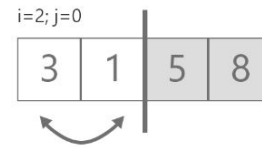
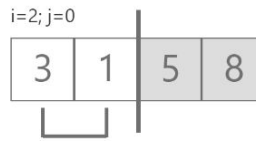
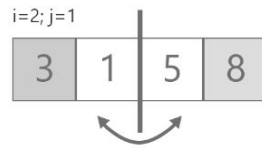
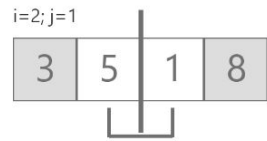
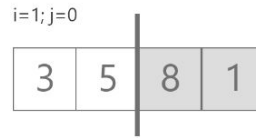
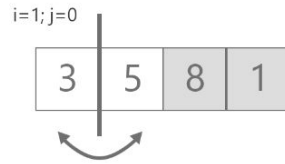
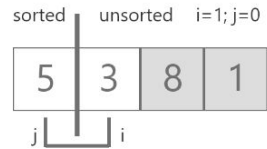
Les tableaux : tri

Tri par insertion

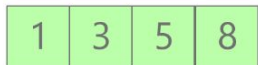
Marche à suivre

- Mettre un marqueur après le premier élément pour séparer les sorted des unsorted
- Boucler en répétant les actions suivantes
 - Sélectionner le premier élément unsorted
 - Déplacer les éléments triés vers la droite pour laisser une place à l'élément unsorted
 - Avancer le marqueur séparant les sorted de 1

Tri par insertion (insertion sort)



résultat final





Les tableaux : tri

Tri par insertion

<https://codepen.io/Snaj/pen/LYOGKBy>



Tri par sélection



Les tableaux : tri

Tri par sélection

$O(n^2)$

On va sélectionner l'élément le plus petit et le placer au début,

On répète sur le tableau en avançant le début de 1



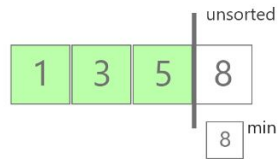
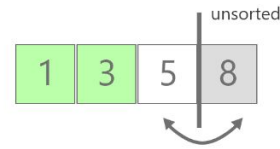
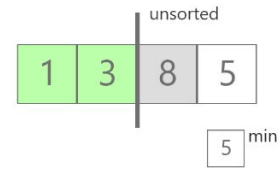
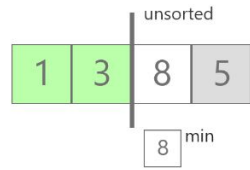
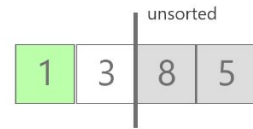
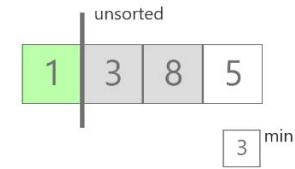
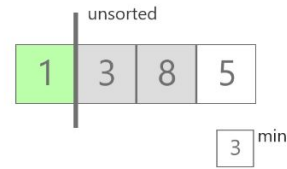
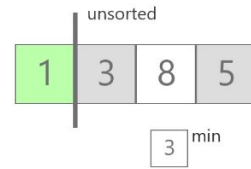
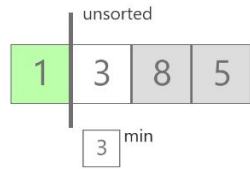
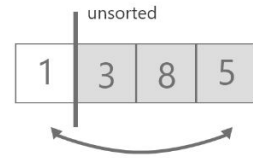
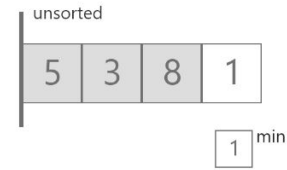
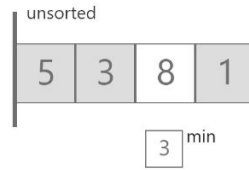
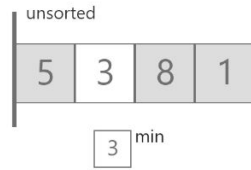
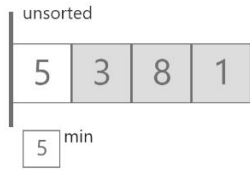
Les tableaux : tri

Tri par sélection

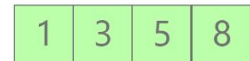
Marche à suivre

- Mettre un marqueur à 0 pour séparer les sorted des unsorted
- Sélectionner le plus petit élément dans la partie unsorted
- L'inverser avec la position du marqueur des sorted
- Avancer le marqueur de 1

Tri par sélection (Selection sort)



résultat final



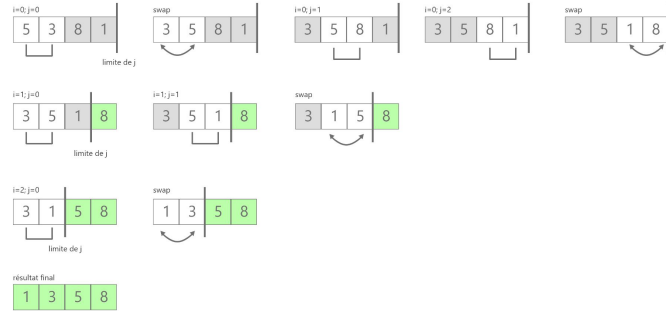


Les tableaux : tri

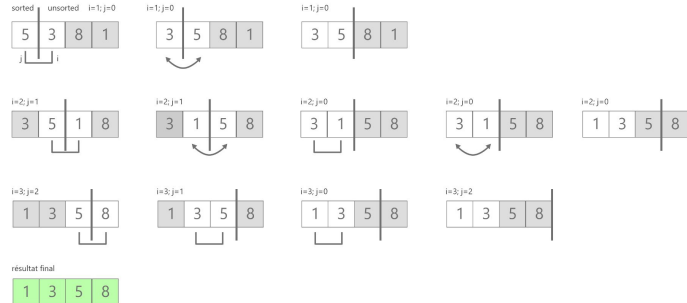
Tri par sélection

<https://codepen.io/Snaj/pen/vYWLoXz>

Tri à bulle (bubble sort)



Tri par insertion (insertion sort)



Tri par sélection (Selection sort)

