

ALGOSTAT 1



etna

LA NOUVELLE ALTERNANCE EN INFORMATIQUE

Rapport de connaissances

*Thomas Joiris
Boris Maigrot
Yoann Rodrigues*

Table des matières

Tris du Sujet	2
Rappel du sujet.....	2
Tri à bulles	2
Tri par sélection	3
Tri par insertion.....	3
Pour aller plus loin	4
Variantes de nos tris.....	4
Autres tris	4

Tris du Sujet

Rappel du sujet

Le but du projet est de comparer et analyser trois algorithmes de tri numérique : le tri à bulles, le tri par sélection et le tri par insertion.

L'idée est de mesurer les performances de ces différents tris en fonction des données à trier et de retourner des mesures temporelles ou des coûts en opérations.

Il faut donc d'abord coder les différents tris en PHP (seul langage autorisé) à partir de l'algorithme en pseudo-code.

Ensuite il faut mettre en place une vue pour interagir avec l'utilisateur dans laquelle il pourra choisir la liste de nombres à trier et un des trois tris du sujet pour effectuer le tri et ses mesures.

Il faut donc aussi coder les différentes fonctions de mesures et de contrôle des données entrées.

Enfin le résultat doit s'afficher sur une page de résultat avec les valeurs retournées par les mesures.

Tri à bulles

Le tri à bulles consiste à faire remonter les éléments les plus grands vers la fin de celui-ci.

Ce tri est lent car on parcourt le tableau ainsi : la première fois on compare les n éléments du tableau avant d'avoir le plus grand en dernier puis on recommence en parcourant $n-1$ éléments (le dernier est à sa place finale) et ainsi de suite jusqu'à arriver à 1 élément à comparer.

Mathématiquement on peut calculer le nombre de comparaisons.

En connaissant la méthode de parcours qui donne :

$$O = n + (n-1) + (n-2) + \dots + 1 = \frac{n(n-1)}{2}$$

$$\text{On a } (n \log n) < \frac{n(n-1)}{2} < n^2$$

Donc on peut en déduire que la complexité du tri à bulles est de $O(n^2)$.

Tri par sélection

Le tri par sélection consiste à chercher le plus petit élément du tableau et le mettre en indice 1 puis le second plus petit et le mettre en indice 2 et ainsi de suite.

Ce tri demande tout comme le tri à bulles de parcourir et comparer tous les éléments du tableau.

On peut donc exactement comme pour le tri à bulles, calculer sa complexité.

Et de fait la complexité de ce tri est également de $O(n^2)$.

Tri par insertion

Le tri par insertion consiste à parcourir un tableau et d'insérer un élément dans un tableau déjà trié.

On va donc parcourir notre tableau et comparer chaque élément avec le dernier élément du tableau trié qui le précède.

Si notre élément est plus grand, on passe au suivant.

Sinon on cherche dans le tableau trié son emplacement.

L'avantage de ce tri est qu'il parcourt et compare tous les éléments entre eux seulement dans le pire des cas (quand la liste est triée à l'envers).

Donc dans tous les autres cas on obtient un nombre d'itérations inférieur aux deux autres tris.

Mathématiquement on peut estimer le nombre d'itérations par $\frac{n(n-1)}{4}$.

On reste quand même sur une complexité de $O(n^2)$.

Pour aller plus loin

Variantes de nos tris

On peut trouver des variantes ou des optimisations des algorithmes de tri que nous avons traités dans ce sujet.

Le tri à bulles possède par exemple 3 variantes connues : tri cocktail, tri jump-down et tri à peigne.

On peut également optimiser le tri à bulles en contrôlant si le tableau est trié à la comparaison de chaque élément et donc prévenir le fait de tourner inutilement si le tableau est trié.

Le tri par insertion possède une variante connue qui est le tri de Shell mais possède de nombreuses optimisations situationnelles.

Autres tris

En travaillant nous sommes forcément tombés sur d'autres algorithmes de tris et notamment de tris plus performant avec une complexité inférieure à ce du sujet.

Le tri à fusion qui consiste à réunir deux listes triées en une seule.

Le tri rapide qui consiste à placer un élément du tableau (appelé pivot) à sa place définitive, en permutant tous les éléments de telle sorte que tous ceux qui sont inférieurs au pivot soient à sa gauche et que tous ceux qui sont supérieurs au pivot soient à sa droite. (Wikipédia)