

ALGO5 : TP sur les tris (2)

Deuxième partie

Statistiques - Récupérer le nombre d'opérations

On s'intéresse aux opérations fondamentales d'un tri : comparaison d'un élément de la liste (avec une autre valeur), et permutations. On va donc compter ces opérations. Pour cela, on va mémoriser dans une structure Stat le nombre de chacune de ces instructions effectuées pour un tri et une taille de liste. Il faut donc avoir un tableau qui permet d'associer à chaque tri (qu'on repérera par son nom), pour chaque taille de liste, le nombre de comparaisons et le nombre de permutations.

Créez votre structure, modifiez votre code et vos signatures de fonction pour ajouter votre objet de statistiques, et remplissez-le en comptant vos opérations.

Questions : Le traitement est-il identique pour tous les tris ? Quelles remarques cela vous inspire-t-il ?

Visualiser les statistiques

Les opérations étant comptées, vous pouvez maintenant afficher les résultats. Une première étape sera de les afficher sur la sortie standard. Une deuxième étape sera d'en faire une sortie dans un fichier au format CSV (*comma separated value*) afin de pouvoir les récupérer dans un tableur et de créer des diagrammes pour visualiser les courbes de progression du nombre d'opérations en fonction de la taille de la liste. Enfin, vous pourrez générer un fichier au format `.dat` pour afficher vos courbes grâce à l'outil `gnuplot`. Vous prévoierez un tableau qui ne prenne en compte que les comparaisons, un tableau qui ne prenne en compte que les permutations et un tableau qui prenne en compte toutes les opérations.

Programmez les fonctions qui permettent une visualisation des résultats :

- sur la sortie standard
- en sortie cvs
- en sortie plot

Discutez les résultats obtenus.

Format CSV : Chaque ligne correspond à une ligne du tableur, et sur une même ligne les colonnes sont séparées par des virgules (on peut utiliser d'autres séparateurs, comme le point-virgule ou la tabulation, mais la virgule est le séparateur habituel).

Format pour GnuPlot : Vous trouverez toutes les informations à partir de la page officielle de GnuPlot <http://gnuplot.info>. GnuPlot est un utilitaire multiplateforme qui permet de tracer des courbes à partir de fonctions ou de fichiers. Il possède un langage de commandes. Le format des fichiers de données pour GnuPlot est très proche du format csv. On trouvera les données sur une même ligne, séparées par des espaces ou des tabulations, les commentaires commencent par un `#`. Exemple (fichier *resultats.dat*) :

Taille de la liste - Nombre d'opérations pour chaque tri

100 5026 5049 7340 1213 1108 1910

200 20139 20099 29716 2825 2625 4414

```
300 45092 45149 66981 4585 4273 7136
400 80253 80199 119371 6449 6044 10024
```

Mais pour obtenir le résultat que nous souhaitons, le mieux est de générer en plus un fichier de commandes qui sera exécuté par GnuPlot. Voici un exemple pour le cas qui nous intéresse (fichier *resultats.gnu*) :

```
# pour avoir un titre sur le diagramme
set title 'Comparaison des algorithmes de tris'

# pour nommer l'axe des abscisses
set xlabel 'Taille de la liste'

# pour nommer l'axe des ordonnées
set ylabel 'Nombre de comparaisons'

# la première courbe est tracée par plot
# en utilisant la colonne 1 pour les abscisses et la colonne 2 pour les ordonnées
plot 'resultats.dat' using 1:2 title 'Tri par insertion' with linespoints

# les courbes suivantes sont dessinées par l'instruction replot
replot 'resultats.dat' using 1:3 title 'Tri par sélection du minimum' with linespoints
replot 'resultats.dat' using 1:4 title 'Tri à bulles' with linespoints
replot 'resultats.dat' using 1:5 title 'Tri fusion' with linespoints
replot 'resultats.dat' using 1:6 title 'Tri rapide' with linespoints
replot 'resultats.dat' using 1:7 title 'Tri par tas' with linespoints

# l'instruction suivante attend que l'utilisateur appuie sur une touche pour quitter l'application
pause -1
```

Il ne reste plus qu'à visualiser les courbes en lançant directement gnuplot avec le fichier de script :

```
moi@mamachine$ gnuplot resultats.gnu
```

Heap sort

Programmez le tri par tas étudié en TD et ajoutez-le à votre plateforme. Tracez les courbes, discutez les résultats.

Tri par tas : Un tas est un arbre binaire parfait dont tous les niveaux sont remplis sauf éventuellement le dernier, qui est rempli à partir de la gauche. Une liste représente un tas où chaque élément d'indice i a pour fils les éléments aux indices $2i$ et $2i+1$. L'algorithme se fait en deux étapes.

1. On crée un tas tel que chaque noeud a une valeur supérieure à celle de ses deux fils. C'est la procédure entasser. Elle parcourt la première moitié de la liste de la droite vers la gauche et permute les valeurs jusqu'à ce que chaque noeud ait une valeur supérieure à celle de ses fils.

2. On bascule plus grand élément du tas (sa racine) à la fin de la liste par permutation, et on replace l'élément qui est arrivé en première position par entassements successifs.

Pour aller plus loin

Il peut être intéressant de regarder le fonctionnement d'un tri dans des situations différentes :

- dans le cas où la liste est déjà presque triée
- dans le cas où elle est déjà triée
- dans le cas où elle est triée dans l'ordre inverse
- dans le cas où elle contient **beaucoup** de doublons

Créez un générateur de listes (pour une taille donnée) pour chacun de ces cas. Adaptez la plateforme pour comparer les différents tris sur les différents types de listes.