

Programmation 2

L1 2011-2012

Devoir à rendre - Tri et Complexité

Le but de ce devoir est de mettre en évidence les différentes complexités de plusieurs algorithmes de tri.

Exercice 1.

Décrire le fonctionnement des algorithmes suivants :

I tri par insertion

B tri à bulle.

E tri par extraction.

R tri rapide (connu aussi sous le nom de Quicksort)

F tri fusion.

Exercice 2.

Dessiner un graphique qui donne le nombre de comparaison moyen des algorithmes en fonction de la taille du tableau. En abscisse figurera la taille des tableaux et en ordonné le nombre de comparaisons moyen. La taille des tableaux ne doit pas dépasser 1000 cases. Les valeurs devront être obtenues sur des exemples de tableaux générés aléatoirement. Les résultats concernant un algorithme doivent être représentés par une courbe. En tout il y a donc 5 courbes à tracer, une par algorithme.

Exercice 3.

Commenter et comparer ces résultats avec les valeurs théoriques attendus. C'est un travail individuel, vous devez rendre un fichier pdf par per-

sonne. Ce devoir est à rendre trois semaines après sa distribution. Sera pris en compte la clarté et la justesse des descriptions et la pertinence des commentaires.

Indication pour la Question 2 :

Écrire pour chaque algorithme une fonction de tri `int triX(int T[], int taille)` qui renvoie le nombre de comparaisons effectuées durant le tri. Pour générer des tableaux aléatoirement vous pouvez utiliser la fonction `random()` définie dans `stdlib.h`. Vous remplissez chaque case de vos tableaux avec une valeur générée par la fonction `random()`. Vous pouvez calculer les valeurs de vos courbes sur des tableaux de taille 10 20 30 40 ... 990 1000. Pour calculer

le nombre de comparaison moyen faite la moyenne du nombre de comparaison sur 10 tableaux (de même taille). Pour tracer le graphique vous pouvez utiliser le programme `gnuplot` : Tapez `gnuplot` dans un terminal pour lancer le programme `gnuplot`. La commande `gnuplot> plot 'monfichier'` trace les points contenus dans le fichier `'monfichier'`. Par exemple si le fichier `'monfichier'` contient

```
0 0
5 4
1 2
2 0.5
3 1.5
```

`gnuplot> plot 'monfichier'` affichera une image sur laquelle il aura tracé les points de coordonnées (0,0), (5,4), (1,2), (2,0.5) (3,1.5).

Pour relier ces points par des lignes vous pouvez utiliser la commande `gnuplot>plot 'monfichier' with lines`

Pour tracer plusieurs courbes vous pouvez utiliser la commande `gnuplot> plot 'monfichier1' with lines,'monfichier2'`

Pour fabriquer une image contenant le graphique (ici png) au lieu l'afficher lancer la commande `gnuplot> set outpout 'monImage.png'` puis `gnuplot> set terminal png`. Il faut lancer ces deux commandes avant la commande `plot`.

Pour quitter `gnuplot` tapez `gnuplot> quit`. Vous pouvez écrire un programme C qui écrit les coordonnées des points dans un fichier, puis ensuite fabriquer l'image grâce à `gnuplot`.