

TRI RAPIDE

TRAVAUX-PRATIQUES D'ALGORITHMIQUE N. 2

Objectifs: Il existe plusieurs versions du tri rapide. La version tri-fusion est de complexité $n \log n$ mais nécessite des structures sophistiquées (liste chaînée) ou bien de l'espace mémoire pour réaliser les opérations de fusions. La version la plus célèbre, QuickSort due C. A. R. Hoare, est particulièrement compacte quadratique dans le pire des cas mais de complexité $n \log n$ en moyenne. Un résultat qui n'est pas vraiment facile à établir. L'objectif de cette séance de travaux pratiques est d'étudier une version pédagogique du tri rapide basée sur une opération de séparation de Nico Lomuto.

Lectures: Algorithmes en Langage C de R. Sedgewick. Algorithm and complexity, S. H. Wilf.

Durée: 3 heures.

```
ALGORITHME TRI-RAPIDE( x, i, j)
DEBUT
  SI ( j > i) ALORS
    k := SEPARER(x, i, j);
    TRI-RAPIDE(x, i, k - 1);
    TRI-RAPIDE(x, k+1, j);
  FSI
FIN
```

FIG. 1. L'algorithme du tri rapide illustre le principe diviser pour régner.

```
PROCEDURE SPLIT( x, g, d )
// separation de Nico Lomuto
DEBUT
  k := g;
  x[k] <-> x[g];
  T := x[g];
  i := g;
  j := i;
  INC( j );
  TANTQUE ( j <= d )
    SI ( X[j] < T ) ALORS
      INC( i );
      x[i] <-> x[j];
    FSI
  INC ( j );
FTQ
  x[g] <-> x[i];
RETOURNER i
FIN
```

Pour les mesures de temps de calculs, utiliser le profileur `gprof` ou encore la librairie `time.h`. Pour les tirages aléatoires, parcourir les pages du manuels des fonctions `rand` et `random`. Attention aux fuites de mémoires, alors, pour éviter un swap désastreux, signalons que la commande `top` permet de voir bouillir la marmite. Pour les représentations graphiques, voir la commande `gnuplot`.

Nous dirons que l'entier x est plus petit ou égal à y ($x \prec y$) si le poids binaire est inférieur de x à celui de y , ou bien, en cas d'égalité, si $x \leq y$. Dans la suite, il s'agit de trier des tableaux d'entiers positifs par rapport à cette relation.

[0] Vérifier que la relation \prec est bien une relation d'ordre. Écrire une fonction `int compare(x, y)` qui renvoie +1, 0, ou -1, analogue de la fonction `strcmp()`.

[1] Implanter le tri rapide fondé sur l'algorithme de séparation de Lomuto.

[2] Tester le bon fonctionnement de votre

[3] Utiliser le tri rapide pour trier des tableaux triés de tailles 500, 1000, 1500, 2000, 2500 etc. . . . Compter les comparaisons. Faire un graphique avec `gnuplot`. Mettre en évidence un équivalent asymptotique.

[4] Introduire de l'aléa dans la procédure de séparation de Lomuto.

[5] Trier des tableaux de tailles 1000, 2000, 3000, 4000, 5000, etc. . . . Compter le nombre d'appels vers la fonction `compare()`. Faire un graphique avec `gnuplot`. Mettre en évidence un équivalent asymptotique.

[6] Faire des mesures de temps de calcul. En déduire le facteur caché de votre implantation.

[7] Utiliser la fonction `qsort()`. Conclusion ?

DÉPARTEMENT D'INFORMATIQUE, UNIVERSITÉ DE TOULON-VAR

Date: Octobre 2003.