TD Tris

CERI - Licence 2 Informatique - Année Académique 2024-2025

13 octobre 2024

Exercice 1.

1/ Faites tourner les algorithmes de tri fusion et rapide (vus en cours) sur la séquence suivante :

$$4, 7, 10, 3, 15, 12, 1, 8, 2, 6$$

2/ La méthode de partitionnement utilisée dans le tri rapide (due à C.A.R Hoare) n'est pas la seule possible. Un autre peut être obtenue comme suit :

```
Algorithme : Partitionnement(T,a,b,m)
i = a − 1 ;
pivot = T[b] ;
pour j allant de a à b − 1 faire
    si T[j] ≤ pivot alors
        i = i + 1 ;
        echanger(T[i], T[j]) ;
    fin
fin
m = i + 1;
echanger(T[i + 1], T[b]) ;
```

Faites tourner le tri rapide sur les mêmes données que précédemment avec ce partitionnement.

3/ On veut montrer que cette façon de faire est valide. C'est à dire qu'elle permet bien de partitionner le tableau dans tous les cas de figure. On dit dans ce cas qu'on veut prouver l'algorithme. La preuve se fait par **récurrence (finie)**.

Au **début** de chaque itération j de la boucle "Pour" on note $i(j)$ le contenu de la variable i. Initialement, quand on entre dans la boucle, $j = a$, $i(j) = a - 1$. L'hypothèse de récurrence (H_j) que l'on veut montrer est la suivante.

Soit j vérifiant $a \leq j \leq b$:
— Si $a \leq k \leq i(j)$ alors $T[k] \leq pivot$.
— Si $i(j) + 1 \leq k \leq j - 1$ alors $T[k] > pivot$
— Si $k = b$ alors $T[k] = pivot$

Montrer par récurrence cette hypothèse, et déduisez-en la validité de l'algorithme.

Correction : Pour bien situer ce que l'on entend par "au début de l'itération", il est indiqué ci-dessous à quel endroit les indices mentionnés dans l'hypothèse s'appliquent.

Algorithme : *Partitionnement(T,a,b,m)*

$i = a - 1$;
$pivot = T[b]$;
pour j *allant de* a *à* $b - 1$ **faire**
    `/* Ici : H_j */`;
    si $T[j] \leq pivot$ **alors**
        $i = i + 1$;
        $echanger(T[i], T[j])$;
    fin
fin
$m = i + 1$;
$echanger(T[i + 1], T[b])$;

La preuve par récurrence d'un algorithme consiste à montrer que l'hypothèse (appelée aussi invariant de boucle) est vraie pour $j = a$. On suppose alors qu'elle l'est pour $a \leq j < b$, et montre qu'elle le reste pour $j + 1$.

La récurrence est "finie" car en pratique on s'arrête à $j = b$ alors qu'une récurrence classique s'applique à un nombre infini de termes d'une suite.

- *Pour $j = a$. $i(j) = a - 1$. Il n'y a donc aucun indice entre a et $i(j)$. Et aucun entre $i(j) + 1$ et $j - 1$. De plus $T[b] = pivot$. L'hypothèse est donc trivialement vérifiée.*

Supposons qu'elle l'est au rang j et montrons qu'elle le reste au rang $j + 1 \leq b$.

Deux cas sont possibles :

$$\begin{array}{ll} (a) & i(j+1) = i(j) + 1 : \text{ quand } T[j] \leq pivot \\ (b) & i(j+1) = i(j) : \text{ sinon} \end{array}$$

- *Cas (a).*

Par hypothèse pour tout k vérifiant $a \leq k \leq i(j)$, $T[k] \leq pivot$. Comme $T[j] \leq pivot$, et du fait de l'échange $(T[i(j) + 1], T[j])$, on a également :

$$Si\ a \leq k \leq i(j+1)\ alors\ T[k] \leq pivot.$$

De même par hypothèse : $i(j) + 1 \leq k \leq j - 1 \Rightarrow T[k] > pivot$.

Donc

$$i(j) + 2 = i(j+1) + 1 \leq k \leq j - 1 \Rightarrow T[k] > pivot.$$

Mais $T[j]$ avait été échangé avec $T[i(j) + 1]$. Comme par hypothèse de récurrence $T[i(j) + 1] > pivot$. On a :

$$i(j+1) + 1 \leq k \leq j \Rightarrow T[k] > pivot$$

- *Cas (b). Trivial car $i(j + 1) = i(j)$.*

Enfin, dans toutes les itérations de la boucle $T[b]$ ne change d'où la validité du 3ème point de l'hypothèse.

On sort de la boucle quand $j = b$. Avec donc :

- *Si $a \leq k \leq i(b)$ alors $T[k] \leq pivot$.*
- *Si $i(b) + 1 \leq k \leq b - 1$ alors $T[k] > pivot$*
- *Si $k = b$ alors $T[k] = pivot$*

L'échange entre $T[i(b) + 1]$ et $T[b]$ permet alors d'avoir tous les éléments inférieurs à $T[b]$ à gauche et tout ceux supérieurs à sa droite.

$\square$