

J1MI2013: Algorithmes et Programmes: feuille 11

Tris récursifs

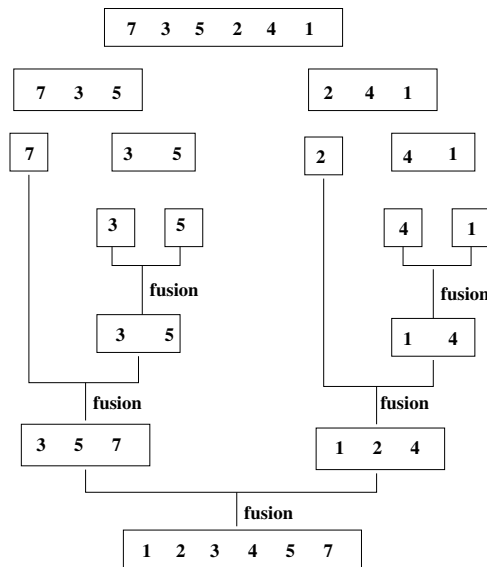
Travaux dirigés

Tri Fusion

Appliquer le tri fusion à un tableau t contenant n éléments consiste à trier récursivement le tableau entre les indices 0 et $\frac{n}{2} - 1$ et entre les indices $\frac{n}{2}$ et $n - 1$, puis à fusionner les deux sous-tableaux triés. L'idée est qu'il est facile de fusionner deux tableaux triés en un tableau lui-même trié.

Exemple. Le tri fusion appliqué au tableau $t = [7, 3, 5, 2, 4, 1]$ le découpe en $[7, 3, 5]$ et $[2, 4, 1]$ qu'on trie eux-mêmes par tri fusion pour obtenir $[3, 5, 7]$ et $[1, 2, 4]$. On fusionne ensuite ces tableaux pour obtenir $[1, 2, 3, 4, 5, 7]$. Ainsi, le tri se déroulera de la façon suivante :

- on considère $[7, 3, 5]$ et on trie récursivement :
 - o on considère $[7]$: ce sous-tableau a un élément, donc est trié.
 - o on considère $[3, 5]$ et on trie récursivement :
 - on considère $[3]$ et $[5]$: chaque sous-tableau a un élément, donc est trié.
 - on fusionne ces deux tableaux et on obtient $[3, 5]$.
 - o on fusionne les deux tableaux $[7]$ et $[3, 5]$: on obtient $[3, 5, 7]$.
- on considère $[2, 4, 1]$ et on trie récursivement :
 - o on considère $[2]$: ce sous-tableau a un élément, donc est trié.
 - o on considère $[4, 1]$ et on trie récursivement :
 - on considère $[4]$ et $[1]$: chaque sous-tableau a un élément, donc est trié,
 - on fusionne ces deux tableaux et on obtient $[1, 4]$.
 - o on fusionne les deux tableaux $[2]$ et $[1, 4]$: on obtient $[1, 2, 4]$
- on fusionne les deux tableaux $[3, 5, 7]$ et $[1, 2, 4]$: on obtient $[1, 2, 3, 4, 5, 7]$



Exercice 1. On rappelle la fonction du TD 7 qui fusionne deux tableaux triés en un troisième :

```
1  int fusion(int t1[], int n1, int t2[], int n2, int t3[]) {
2      int i1 = 0;
3      int i2 = 0;
4      while (i1 < n1 && i2 < n2) {
5          if (t1[i1] < t2[i2]){
6              t3[i1+i2] = t1[i1];
7              i1++;
8          } else {
9              t3[i1+i2] = t2[i2];
10             i2++;
11         }
12     }
13     while (i1 < n1) {
14         t3[i1+i2] = t1[i1];
15         i1++;
16     }
17     while (i2 < n2) {
18         t3[i1+i2] = t2[i2];
19         i2++;
20     }
21     return n1 + n2;
22 }
```

En s'inspirant de cette fonction, écrire une fonction `fusionner(t,debut, milieu, fin)` qui, étant donné un tableau `t` trié entre les indices `debut` et `milieu-1` et entre les indices `milieu` et `fin-1`, trie `t` entre les indices `debut` et `fin-1`.

Par exemple, soit `t=[4, 3, 6, 1, 5, 7]`. On remarque que `[3, 6]` est trié donc `t` est trié entre les indices 1 et 2. De la même façon, `[1, 5]` est trié donc `t` est trié entre les indices 3 et 4. La fonction `fusionner(t,1,3,5)` va ainsi fusionner ces deux sous-tableaux et on aura `t=[4, 1, 3, 5, 6, 7]` qui est trié entre les indices 1 et 4.

Exercice 2. Écrire une fonction récursive `triFusionRec(t,debut,fin)` qui trie récursivement le tableau `t` entre les indices `debut` et `fin-1`, en utilisant la fonction `fusionner` écrite à l'exercice 1.

Exercice 3. Écrire une fonction `triFusion(t, n)` qui trie le tableau `t` de taille `n` en utilisant `triFusionRec`.

Tri Rapide

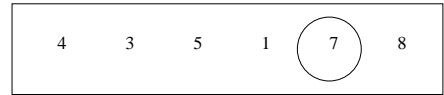
L'idée du tri rapide est de choisir dans le tableau à trier un élément qu'on appelle le pivot (on peut par exemple prendre le premier élément du tableau) et de mettre à sa gauche tous les éléments plus petits que lui et à sa droite tous les éléments plus grands que lui. Pour trier la tableau initial, il ne reste alors plus qu'à trier les sous-tableaux à gauche et à droite du pivot, ce qu'on fait en itérant récursivement le tri rapide.

Exemple : soit $t = [7, 3, 5, 1, 8, 4]$ le tableau à trier :

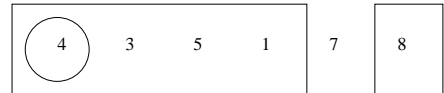
On considère le tableau $[7, 3, 5, 1, 8, 4]$. Le pivot est 7. On réorganise et on obtient $[4, 3, 5, 1, 7, 8]$. Il faut donc trier les tableaux $[4, 3, 5, 1]$ et $[8]$.



– On considère le tableau $[4, 3, 5, 1]$. Le pivot est 4. On réorganise et on obtient $[1, 3, 4, 5]$. Il faut donc trier les tableaux $[1, 3]$ et $[5]$



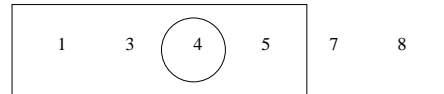
◦ On considère $[1, 3]$. Le pivot est 1. On réorganise et on obtient $[1, 3]$. Il faut trier les tableaux $[]$ et $[3]$.



— Le tableau $[]$ est trié.

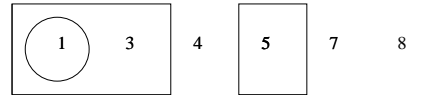
— Le tableau $[3]$ est trié.

Le tableau $[1, 3]$ est donc trié.



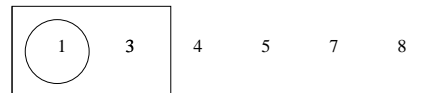
◦ Le tableau $[5]$ est trié.

Le tableau $[1, 3, 4, 5]$ est donc trié.

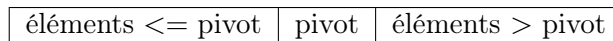


– Le tableau $[8]$ est trié

Le tableau $[1, 3, 4, 5, 7, 8]$ est donc trié



Exercice 4. Écrire une fonction `reorganiser(t,debut,fin)` qui utilise comme pivot l'élément d'indice `debut` et réorganise le sous-tableau situé entre les indices `debut` et `fin-1` de manière que l'on obtienne :



Cette fonction retourne l'indice de la nouvelle position du pivot.

Exercice 5. Écrire une fonction récursive `triRapideRec(t,debut,fin)` qui trie récursivement le tableau `t` entre les indices `debut` et `fin`.

Exercice 6. Écrire une fonction `triRapide(t,n)` qui trie le tableau `t` de taille `n` en utilisant `triRapideRec`.