

4TPM205U: Algorithmique des tableaux: feuille 9

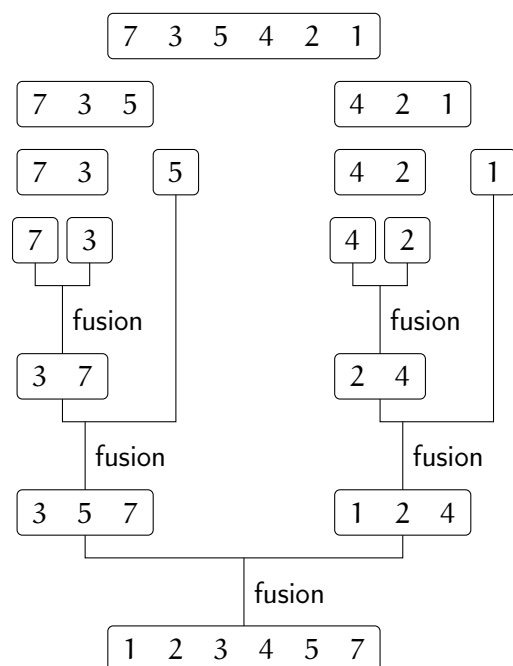
Tris récursifs : tri fusion

Travaux dirigés

Appliquer le tri fusion à un tableau t contenant n éléments consiste à trier récursivement le tableau entre les indices 0 et $\lfloor \frac{n-1}{2} \rfloor$ et entre les indices $\lfloor \frac{n-1}{2} \rfloor + 1$ 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 $tab = [7, 3, 5, 4, 2, 1]$ le découpe en $[7, 3, 5]$ et $[4, 2, 1]$ 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 :
 - on considère $[7, 3]$ et on trie récursivement :
 - on considère $[7]$ et $[3]$: chaque sous-tableau a un élément, donc est trié.
 - on fusionne ces deux tableaux et on obtient $[3, 7]$.
 - on considère $[5]$: ce sous-tableau a un élément, donc est trié.
 - on fusionne les deux tableaux $[3, 7]$ et $[5]$: on obtient $[3, 5, 7]$.
- on considère $[4, 2, 1]$ et on trie récursivement :
 - on considère $[4, 2]$ et on trie récursivement :
 - on considère $[4]$ et $[2]$: chaque sous-tableau a un élément, donc est trié,
 - on fusionne ces deux tableaux et on obtient $[2, 4]$.
 - on considère $[1]$: ce sous-tableau a un élément, donc est trié.
 - on fusionne les deux tableaux $[2, 4]$ et $[1]$: on obtient $[1, 2, 4]$
- on fusionne les deux tableaux $[3, 5, 7]$ et $[1, 2, 4]$: on obtient $[1, 2, 3, 4, 5, 7]$



On rappelle la fonction du TD #5 qui fusionne deux tableaux triés en un troisième :

```
1 def fusion (t1, n1, t2, n2, t3):
2     i1 = 0
3     i2 = 0
4     while i1 < n1 and i2 < n2:
5         if t1[i1] < t2[i2]:
6             t3[i1 + i2] = t1[i1]
7             i1 += 1
8         else:
9             t3[i1 + i2] = t2[i2]
10            i2 += 1
11    while (i1 < n1):
12        t3[i1 + i2] = t1[i1]
13        i1 += 1
14    while (i2 < n2):
15        t3[i1 + i2] = t2[i2]
16        i2 += 1
17    return n1+n2
```

Exercice 1

1. Si on applique le `triFusion` à un tableau de n éléments, où se trouve le premier couple d'éléments qui seront comparés entre eux? Où se trouve le deuxième couple d'éléments qui seront comparés entre eux? Qu'est-ce qu'on peut dire sur les indices respectifs du dernier couple d'éléments qui seront comparés entre eux?
2. Donner le nombre de comparaisons nécessaires entre éléments du tableau pour effectuer le tri fusion d'un tableau de taille n dans le pire des cas (quel est-il?), puis dans le meilleur des cas (quel est-il?).
3. Donner le nombre d'affectations à une case du tableau nécessaires pour effectuer le tri fusion d'un tableau de taille n dans le pire des cas (quel est-il?), puis dans le meilleur des cas (quel est-il?).
4. En fonction du nombre n d'éléments à trier, quel est l'ordre de grandeur de la complexité en temps du tri fusion?
5. Combien de fois au maximum déplace-t-on un élément du tableau?
6. Si le tableau contient deux occurrences du même élément, est-ce qu'après le tri l'ordre initial des deux occurrences entre elles est encore respecté?

Exercice 2

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

Par exemple, soit `t=[4, 3, 6, 2, 5, 1]`. On remarque que `[3, 6]` est trié donc `t` est trié entre les indices 1 et 2. Pareillement, `[2, 5]` est trié donc `t` est trié entre les indices 3 et 4.

L'appel de la fonction : `fusionner(t, 1, 2, 4)` va ainsi fusionner ces deux sous-tableaux et on aura `t=[4, 2, 3, 5, 6, 1]` qui est trié entre les indices 1 et 4.