

4TPM205U: Algorithmique des tableaux: feuille 9

Récurtivité (2)

Travaux dirigés sur machine

Rappel : le fichier `bibTableau.py` doit se trouver dans le même répertoire que vos programmes Python.

Exercice 1 – Tri fusion

1. Écrire une fonction récursive `triFusionRec(t,debut,fin)` qui trie récursivement le tableau `t` entre les indices `debut` et `fin`, en utilisant la fonction `fusionner` définie dans l'exercice 2 de la feuille de TD #9.
2. Écrire une fonction `triFusion(t, n)` qui trie le tableau `t` de taille `n` en utilisant `triFusionRec`.

Exercice 2 – Expérimentation

Lancer l'application <http://inriamecsci.github.io/#!/grains/methodes-tri>

1. Testez le tri fusion en visuel. A l'aide de l'onglet `tri temporel` faites des tests sur le temps de calcul pour trier par exemple :
 - 100 tableaux de taille 100
 - 100 tableaux de taille 200
 - 100 tableaux de taille 500
 - 500 tableaux de taille 500
 - 1000 tableaux de taille 500
 - 1 tableau de taille 3000
2. Comparez les résultats stockés dans le journal avec les résultats théoriques vus en cours et en TD.

Exercice 3 Écrire une version récursive des fonctions `quotient`, `reste` et `sommeChiffres` pour lesquelles on a utilisé des algorithmes itératifs dans la feuille de TD #2.

Exercice 4 Écrire une fonction récursive `nbOccRec(t,n,elt)` qui calcule et renvoie le nombre d'occurrences de l'élément `elt` parmi les `n` premiers éléments du tableau `t`, $n \geq 0$.

Exercice 5 Écrire une fonction récursive `maximumRec(t,n)` qui calcule et renvoie le maximum parmi les `n` premiers éléments du tableau `t` non vide ($n \geq 1$).

Exercice complémentaire 1 – On désire maintenant utiliser pour la fusion un algorithme proche de celui étudié pour insérer un tableau trié dans un tableau trié (voir l'exercice 1 du TD machine #5). Réécrire les fonctions `fusionner`, `triFusionRec` et `triFusion` de manière à intégrer cette modification.