

Algorithmique et programmation procédurale - TD No 4

Tableaux

Exercice 1

Écrire une fonction qui retourne toutes les valeurs \geq le premier élément dans un tableau.

Exemple : `plusGrandQuePremier ({7,4,6,8,3,7,4,9}) = {7,8,7,9}`

Exercice 2

Écrire une fonction qui vérifie si les éléments d'un tableau donné sont triés ou pas (de façon croissante ou décroissante !).

Exercice 3

Écrire une fonction qui reçoit un tableau trié et une valeur, puis « insère » cette valeur (renvoie un nouveau tableau avec une case de plus). Évaluer le nombre d'affectations nécessaires à votre fonction dans le meilleur et le pire des cas, et en déduire sa complexité.

Exercice 4

Écrire un algorithme qui permet de trouver l'intersection de deux tableaux triés dans l'ordre strictement croissant. Par exemple, l'intersection de $\{1, 3, 4, 6, 8\}$ et de $\{4, 5, 6, 12\}$ donne $\{4, 6\}$. Si les longueurs respectives de deux tableaux sont n et m , la complexité de votre algorithme doit être $O(n+m)$ et non $O(n*m)$.

Exercice 5

Proposer une version itérative de l'algorithme de recherche dichotomique dans un tableau.

Exercice 6. Tri par insertion

Réécrire le tri par insertion vu en cours pour trier dans l'ordre décroissant. Est-il possible d'améliorer le coût de cet algorithme en appliquant une recherche dichotomique pour l'insertion d'une valeur dans le sous tableau déjà trié ?