Université Paris Cité

L2 informatique - Année 2023–2024

TD d'Éléments d'Algorithmique n° 5

Dans ce TD, "trié" signifie "trié par ordre croissant".
* Les exercices marqués d'une étoile sont à faire à la maison.

Exercice 1. *Dichotomie.*

1. Exécutez l'algorithme de dichotomie récursif vu en cours sur la valeur 42 et chacun des tableaux suivants :

$$[1, 2, 42, 57, 99] \qquad [1, 2, 3, 4, 42] \qquad [1, 2, 3, 57, 99]$$

   Combien de comparaisons a-t-il fallu faire dans chaque cas ?

2. Même question avec l'algorithme itératif.

Exercice 2. *Recherche dans un tableau arbitraire.*

Dans cet exercice, on suppose qu'un test d'égalité qui implique des éléments d'un tableau a le même coût qu'une comparaison qui implique des éléments du tableau et on appelle les deux opérations des "comparaisons". On a un tableau de taille n dans lequel on veut effectuer m recherches. Combien de comparaisons faut-il faire dans le pire des cas :

1. si on effectue m recherches séquentielles ;
2. si on effectue un tri par insertion suivi de m recherches par dichotomie.
3. si on effectue un tri faisant $n \log n$ comparaisons (par exemple le tri fusion) suivi de m recherches par dichotomie ?

Exercice 3. *Diviser pour régner.*

On dispose d'un tas de n pièces $p_0, p_1 \dots p_{n-1}$ dont exactement une est fausse ; toutes les pièces ont le même poids sauf la pièce fausse, qui est plus légère. On dispose d'une balance à deux plateaux.

1. Combien de pesées sont nécessaires pour déterminer la pièce fausse lorsque $n = 4$? $n = 8$? $n = 9$?
2. Écrivez un algorithme récursif qui permet de déterminer la pièce fausse lorsque le nombre de pièces est une puissance de deux. Combien de pesées fait-il ?
3. Même question lorsque le nombre de pièces est une puissance de trois.
4. Et dans le cas général ?

Exercice 4. *Recherche dans un tableau bi-dimensionel.*

Dans cet exercice, on considère un tableau bi-dimensionel T de taille $n \times m$ d'entiers dont les lignes et les colonnes sont triées. On veut effectuer une recherche d'un élément x dans ce tableau.

1. Écrivez un algorithme qui fait une recherche séquentielle. Combien de comparaisons effectue votre algorithme au pire de cas ?
2. Écrivez un algorithme qui fait n recherches dichotomiques (une par ligne). Combien de comparaisons effectue cet algorithme au pire de cas ?
3. Écrivez un algorithme qui fait m recherches dichotomiques (une par colonne). Combien de comparaisons effectue l'algorithme dans ce cas ?

4. Enfin, proposer un algorithme *efficace* qui n'effectue qu'au plus $m + n$ comparaisons au pire de cas.

5. Exécuter les algorithmes proposés pour l'élément $x = 14$ sur le tableau T suivant :

1	4	6	7	9
2	7	8	9	11
3	9	10	13	15
6	10	15	17	20
14	18	19	21	22

Exercice 5. *Point fixe.*

On considère un tableau trié T d'entiers relatifs tous distincts. On dit qu'un indice i est un *point fixe* de T si $T[i] = i$, un *pré-point fixe* si $T[i] \leq i$, et *post-point fixe* si $T[i] \geq i$.

1. Quels sont les pré- et post-points fixes du tableau $T = [-1, 0, 1, 3, 4, 8]$?

2. Quelle propriété satisfont les ensembles de pré- et post-points fixes d'un tel tableau ?

*3. Concevez un algorithme itératif qui prend en entrée un tableau trié T d'entiers relatifs tous distincts et retourne vrai si et seulement s'il existe un indice i tel que $T[i] = i$. Combien votre algorithme fait-il de comparaisons dans le cas pire ? Esseyez de rendre votre algorithme aussi efficace que possible.