

IFT2015 Hiver 2010 — Devoir 3

Miklós Csűrös

17 mars 2010

À remettre lors avant 16 :30 le 31 mars. Ce travail est destiné à des équipes de deux ou trois étudiant/es. Remettez votre travail en format PDF par courriel (à csuros@iro. . .).

1 Exercices simples (15 points)

a. (5 points) Dessiner des arbres binaires de recherche de hauteur 2, 3, 4, 5, et 6 pour le même ensemble de clés $\{1, 4, 5, 10, 16, 17, 21\}$.

b. (5 points) Dessiner l'arbre binaire de recherche complet de hauteur 3 à partir des clés $\{1, 2, \dots, 15\}$. Supposons que c'est un arbre *splay* à un moment donné. Montrer le résultat du déploiement après chaque opération dans la séquence

`search(2), search(10), delete(4).`

c. (5 points) On suppose que les entiers compris entre 1 et 1000 sont disposés dans un arbre binaire de recherche, et on souhaite retrouver le nombre 363. Parmi les séquences suivantes, lesquelles ne pourraient pas être la séquence de nœuds parcourus? Justifiez votre réponse.

a. 2, 252, 401, 398, 330, 344, 397, 363.

b. 924, 220, 911, 244, 898, 258, 362, 363.

c. 925, 202, 911, 240, 912, 245, 363.

d. 2, 399, 387, 219, 266, 382, 381, 278, 363.

e. 935, 278, 347, 621, 299, 392, 358, 363.

2 Problèmes (45+15 points)

Faites **trois des quatre** exercices suivants. Vous pouvez soumettre une solution du quatrième aussi pour 15 points de boni.

2.1 Séquence bitonique

Le tableau $A[0 \dots n-1]$ est appelé **bitonique** s'il existe un indice $i \in \{0, 1, \dots, n-1\}$ tel que

$$A[0] \leq A[1] \leq \dots \leq A[i]; \text{ et} \\ A[i] \geq A[i+1] \geq A[i+2] \geq \dots \geq A[n-1].$$

Donnez un algorithme pour trier A dans l'ordre croissant en temps $O(n)$ au pire. [**Indice** : examinez la technique du tri par fusion.]

2.2 Vis-écrou

On a un ensemble de n vis et n écrous. On sait que les vis sont de tailles différentes, et qu'il existe exactement un écrou pour chaque vis. Donnez un algorithme pour trouver l'écrou à chaque vis avec $O(n \log n)$ comparaisons *en moyenne*. On ne peut comparer qu'un écrou E à une vis V à la fois (donc pas de comparaisons écrou-écrou ou vis-vis), pour vérifier que soit $E < V$, soit $E = V$, soit $E > V$. [**Indice** : examinez la technique du tri rapide.]

2.3 Les tout petits

On a un min-tas de taille n stocké dans le tableau $T[0 \dots n-1]$. Donnez un algorithme qui, pour un paramètre k donné, énumère les indices $i \in \{0, 1, \dots, n-1\}$ avec $T[i] \leq k$ (dans un ordre quelconque). L'algorithme doit prendre $O(m)$ temps pour une réponse comprenant m indices (donc $m = |\{i: T[i] \leq k\}|$).

2.4 Nombre manquant

Soit T un arbre binaire de recherche complet de hauteur k avec clés entières dans l'ensemble $\{1, 2, \dots, 2^{k+1}\}$. Comme T contient exactement $2^{k+1} - 1$ nœuds [non-nulls], il y a un $i \in \{1, \dots, 2^{k+1}\}$ qui n'est pas une clé dans T . Donnez un algorithme pour trouver cette clé en temps $O(k)$.