

IFT2015 automne 2013 — Devoir 4

Miklós Csűrös

4 novembre 2013 — remise mercredi le 13 octobre à 20 :15

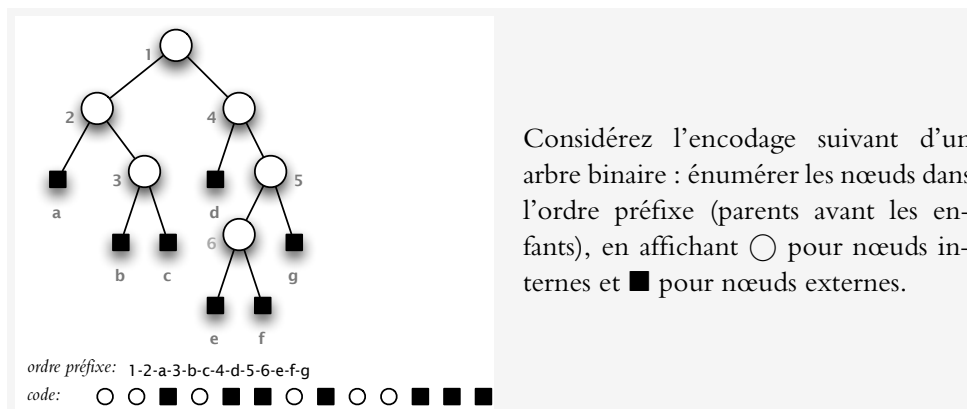
Remettez un rapport écrit par email (à csuros@iro...) en format PDF. La taille de votre message ne doit pas dépasser 2^{20} octets. Travaillez seul sur tous les problèmes. Le devoir vaut 50 points.

4.1 Tas médiane (25 points)

La *médiane* d'une collection d'éléments comparables $x_1 \leq x_2 \leq \dots \leq x_n$ est l'élément «au milieu» $x_{\lceil n/2 \rceil}$. Exemples : la médiane de $\{1, 2, 3, 4\}$ est 2 ; la médiane de $\{5, 6, 7\}$ est 6 ; et la médiane de $\{8, 8, 8, 9, 9\}$ est 8. ► Concevoir une structure de données qui implante les opérations `insert(x)` et `deleteMedian()`. L'opération `insert` insère un nouvel élément ; `deleteMedian` supprime et retourne la médiane. Toutes les deux opérations doivent s'exécuter en $O(\log n)$.

Indice: Utiliser un min-tas et un max-tas.

4.2 Encodage d'arbres (25 points)



Considérez l'encodage suivant d'un arbre binaire : énumérer les nœuds dans l'ordre préfixe (parents avant les enfants), en affichant ○ pour nœuds internes et ■ pour nœuds externes.

Décodage (17 points). ► Donner un algorithme qui construit un arbre binaire à partir de son encodage. Vous pouvez assumer que l'entrée, un tableau $C[0..l-1]$, est valide. L'algorithme doit construire l'arbre et retourner sa racine. Chaque nœud interne N possède les variables $N.parent$, $N.left$, $N.right$, pour le parent, et les enfants gauche et droit. Les nœuds externes sont null.

Indice: Utiliser une méthode récursive auxiliaire qui prend le tableau C et un indice i , et construit le sous-arbre complète encodé par le segment qui débute avec indice i . La méthode peut retourner une paire (j, N) où j est l'indice après ($j = i + \text{nombre de nœuds dans le sous-arbre}$) et N est la racine de sous-arbre.

Nombre d'arbres binaires (8 points) Soit t_n le nombre d'arbre binaires à n nœuds internes. ► Démontrer que $t_n < \binom{2n}{n} < 4^n$ pour tout $n > 0$.

Indice: Utilisez l'encodage des arbres et les propriétés de l'encodage dans l'argument.

REMARQUE. On a $t_0 = 1, t_1 = 1, t_2 = 2, t_3 = 5, t_4 = 14, \dots$. On peut compter le nombre d'arbres binaires selon la taille des sous-arbres de la racine. Il y a exactement $t_k \cdot t_{n-1-k}$ arbres binaires avec k nœuds internes dans le sous-arbre gauche de la racine (et $(n-1-k)$ nœuds internes dans le sous-arbre droit). En conséquence, le nombre d'arbre binaires se donne par la récurrence

$$t_n = \begin{cases} 1 & \{n = 0\} \\ \sum_{k=0}^{n-1} t_k \cdot t_{n-1-k} & \{n > 0\} \end{cases}$$

La solution est $t_n = \frac{1}{n+1} \binom{2n}{n}$. Par l'approximation Stirling, $t_n \sim \frac{4^n}{\sqrt{\pi n^3}}$.