

# IFT2015 automne 2011 — Devoir 3

Miklós Csűrös

12 octobre 2011

À remettre avant 20 :15 le 19 octobre par email (à csuros@iro...).

## 3.1 Le nœud moyen (12 points)

Soit  $d[x]$  la profondeur (*depth*) et  $h[x]$  la hauteur (*height*) d'un nœud  $x$ . On s'intéresse à la moyenne de  $d$  et  $h$  parmi les nœuds internes (dénotées par  $\bar{d}$  et  $\bar{h}$ ) quand leur nombre  $n$  va vers l'infini : est-ce qu'elles sont de même ordre ?

**a. (3 points)** ► Démontrez que  $\bar{d} + \bar{h} \leq H$  pour tout arbre de hauteur  $H$ .

**b. (9 points)** Examinez les «suspects de convenance» : un arbre binaire complet avec  $n = 2^h - 1$  nœuds internes et un arbre linéaire avec  $n = h$  nœuds internes.

► Calculez  $\bar{d}$  et  $\bar{h}$  exactement pour les deux types d'arbres comme fonctions de  $n$ .

► Caractériser la croissance des moyennes en notation asymptotique dans ces deux cas. **Indice.** Pour travailler avec des sommes de genre  $S = \sum_i iq^i$ , utilisez l'astuce de comparer  $qS$  ou  $S/q$  avec  $S$ .

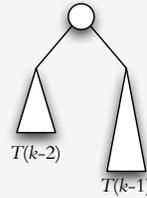
## 3.2 Compilateur polonais (12 points)

► Donnez un algorithme pour construire un arbre syntaxique pour expressions arithmétiques à partir de la notation préfixe. L'entrée est une séquence  $x_1, x_2, \dots, x_n$  où chaque  $x_i$  est une constante ou une opération (à deux arguments). L'arbre syntaxique est un arbre binaire où les nœuds internes sont des opérations, et les nœuds externes sont des constantes. L'algorithme doit calculer les références aux enfants gauche et droit (champs  $u.left$  et  $u.right$ ) pour chaque nœud interne  $u$  et retourner la racine de l'arbre. (Vous pouvez assumer que la séquence est de syntaxe correcte.) ► Montrez l'arbre syntaxique pour l'expression  $+ * 4 + 5 3 - 1 7$ . **Indice.** Utilisez la récursion pour construire les nœuds de l'arbre.

### 3.3 Arbre Fibonacci (6 points)



On définit les arbres binaires Fibonacci  $T(k)$  :  $k = 1, 2, \dots$  par récursion. Les arbres  $T(1)$  et  $T(2)$  ont un seul nœud externe. Pour  $k > 2$ , les deux enfants de la racine de  $T(k)$  sont  $T(k-1)$  et  $T(k-2)$ .



Pour chaque arbre  $T(k)$ , soit  $h(k)$  sa hauteur et  $n(k)$  le nombre de ses nœuds internes. ► Démontrez que  $h(k) = \Theta(\log n(k))$ . **Indice.** Vous pouvez se servir du fait que  $f = \Theta(g)$  si  $0 < \lim_{i \rightarrow \infty} f(i)/g(i) < \infty$  (quand cette limite existe).