Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr
Tobias Lieber

Wintersemester 2010/11
Übungsblatt 3
9. November 2010

# Effiziente Algorithmen und Datenstrukturen

*Abgabetermin: 16. November 2010 vor der Vorlesung*

## Aufgabe 1 (10 Punkte)

Geben Sie Algorithmen `postNext(v)` und `preNext(v)` an, die zu einem Knoten $v$ in einem Binärbaum den in der PreOrder bzw. PostOrder folgenden Knoten $w$ berechnet. Analysieren Sie die asymptotische Worst-Case Laufzeit Ihres Pseudocodes.
Berechnen Sie außerdem die asymptotische Laufzeit wenn mittels der Operationen `postNext(v)` und `preNext(v)` die vollständige PreOrder bzw. PostOrder berechnet wird (also $n$-maliges Anwenden der Funktion).

## Aufgabe 2 (10 Punkte)

Sei $T$ ein Binärbaum, so dass jeder Knoten 0 oder 2 Kinder hat, und sei weiterhin $v$ ein innerer Knoten von $T$ mit linkem Kind $v_l$ und rechtem Kind $v_r$, dann heißt

$$\rho(v) = \frac{w(v_l)}{w(v)} = 1 - \frac{w(v_r)}{w(v)}$$

die Balance von $v$, wobei das Gewicht $w(x)$ die Anzahl der Blätter im Unterbaum mit Wurzel $x$ ist. Ein Binärbaum $T$ heißt $\alpha$-balanciert, für $0 < \alpha < 1$, wenn für jeden inneren Knoten $v$ von $T$ gilt

$$\alpha \leq \rho(v) \leq 1 - \alpha \ .$$

Zeigen Sie, dass für die Höhe $h(T)$ jedes $\alpha$-balancierten Binärbaumes $T$ mit $n$ inneren Knoten gilt

$$h(T) \leq 1 + \frac{\log(n+1) - 1}{\log(1/(1-\alpha))} \ .$$

## Aufgabe 3 (10 Punkte)

(a) Zeigen Sie, dass die Tiefe eines Rot-Schwarz-Baumes mit $n$ Blättern $O(\log n)$ ist.

(b) Beschreiben Sie die Delete-Operation in Rot-Schwarz-Bäumen. Achten Sie hierbei darauf, dass der Baum auch nach der Operation ein Binärbaum sein muss. Zeigen Sie, dass die Laufzeit einer Delete-Operation $O(\log n)$ ist.

*Hinweis:* Durch geeignete Fallunterscheidung kann der Baum so traversiert und modifiziert werden (vom zu entfernenden Blatt bis maximal zur Wurzel und wieder zurück), dass das Blatt mittels eines elementaren Schrittes entfernt werden kann.

## Aufgabe 4 (10 Punkte)

Fügen Sie die Schlüssel $a, b, \dots, i$ in der Reihenfolge $(a, i, e, g, h, f, c, b, d)$ in einen anfänglich leeren AVL-Baum ein. Entfernen Sie anschließend den Schlüssel $i$. Zeichnen Sie jeweils den resultierenden Baum (einschließlich notwendiger Rotationen).