Technische Universität München
Institut für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr

WS 2002/03
Übungsblatt 2
25. Oktober 2002

# Effiziente Algorithmen und Datenstrukturen I

*Letzter Abgabetermin: Montag, 4. November 2002 (vor der Übung)*

## Aufgabe 1

Geben Sie die explizite Lösung für folgende Rekursionsgleichung an:

$$\begin{aligned} a_0 &= a_1 = 0 \\ a_n &= 5a_{n-1} - 6a_{n-2} + n + 1, \qquad \text{für } n \geq 2 \end{aligned}$$

Berechnen Sie die Lösung mittels der erzeugenden Funktion $A(z) = \sum_{n\geq 0} a_n z^n$.

*Hinweis*: Im Falle von Komplikationen beim Berechnen der geschlossenen Lösung kann auf das Zwischenergebnis $A(z) = \frac{3z^2-2z^3}{(1-z)^2(1-3z)(1-2z)}$ zurückgegriffen werden.

## Aufgabe 2

Ein binärer Suchbaum ist ein Baum, in dem jedem Knoten ein Schlüsselwert zugeordnet ist. In dieser Aufgabe betrachten wir als Menge der Schlüssel die ganzen Zahlen $\mathbb{Z}$. Für jeden inneren Knoten $v$ gilt, dass alle in seinem linken bzw. rechten Unterbaum (soweit existent) gespeicherten Schlüssel kleiner bzw. größer als der Schlüssel von $v$ sind. Ferner soll gelten, dass jeder Schlüssel höchstens einmal im Baum gespeichert ist.

a) Beschreiben Sie eine Funktion $\textsc{Next}(T, i)$, die zu einer gegebenen Zahl $i \in \mathbb{Z}$ den nächst größeren im Baum $T$ gespeicherten Schlüssel ausgibt. Existiert kein solcher Schlüssel, soll der Wert UNDEF zurückgegeben werden.

b) Beschreiben Sie eine Prozedur $\textsc{Delete}(T, i)$, die zu einer gegebenen Zahl $i \in \mathbb{Z}$ testet, ob diese als Schlüssel im Baum $T$ vorkommt und ggf. den entsprechenden Knoten löscht. Hierbei soll, falls nötig, der verbleibende Baum so umgeformt werden, dass wieder ein gültiger Suchbaum entsteht.

c) Geben Sie die Laufzeit der beiden Algorithmen aus a) und b) jeweils in Abhängigkeit von der Anzahl $n$ der Schlüssel im Baum sowie der Tiefe $d$ des Baums an.

*Hinweis*: Wird in den Teilaufgaben a) und b) die Beschreibung mittels eines Programms in Pseudo-Code angegeben, ist dieses entsprechend zu kommentieren. Programme ohne Kommentar werden nicht akzeptiert.

## Aufgabe 3

Zeigen Sie, dass es keine Konstante $c > 1$ gibt, so dass gilt:

> Sei $T$ ein $(2,4)$-Baum mit $n$ Blättern, und sei $v$ ein Knoten in $T$ der Tiefe $d$. Dann ist die Anzahl der Blätter in $v$'s Unterbaum von oben (bzw. von unten) durch $n/c^d$ beschränkt.

*Bemerkung:* Im Unterschied zu gewichtsbalancierten Bäumen sind die Blätter in $(2,4)$-Bäumen **nicht** gleichmäßig verteilt.

## Aufgabe 4

Es seien $T_1$ und $T_2$ zwei $(a,b)$-Bäume mit $n_1$ bzw. $n_2$ Knoten, so dass für alle $x \in T_1$ und $y \in T_2$ gilt: $\text{key}(x) < \text{key}(y)$.
Entwerfen Sie eine Prozedur CONCATENATE, die $T_1$ und $T_2$ zu einem neuen $(a,b)$-Baum verschmilzt und deren Laufzeit $\mathrm{O}(\log(n_1 + n_2))$ ist.