INSTITUT FÜR INFORMATIK V
UNIVERSITÄT BONN

Römerstraße 164
53117 Bonn

Prof. Dr. Norbert Blum
Matthias Kretschmer

05. November 2007

Informatik III
Algorithmen und Berechnungskomplexität I
WS 2007/08
Übungen – Blatt 3

Abgabe: 12.11.2007 bis 11Uhr vor HS D

Aufgabe 1 (15 Punkte)**.** Zeigen Sie, dass die Anzahl der benötigten Rotationen bei einer Streicheoperation in einem AVL-Baum mit n Knoten $\Omega(\log n)$ beträgt (worst-case).

Hinweis: Konstruieren Sie einen AVL-Baum mit n Knoten, bei dem möglichst viele Rotationen durchgeführt werden müssen.

Aufgabe 2 (30 Punkte)**.** Seien $k \in \mathbb{N}_0$ und S eine sortierte Menge. Dann ist die Operation SucheUndNachfolger(a, k, S) wie folgt definiert:

- Falls $a \in S$: Gebe a und die nächsten k größeren Elemente aus S aus.
- Falls $a \notin S$: Gebe „$a \notin S$“ aus.

Beispiel: Bestehe S aus den Elementen 1, 2, 10, 12, 50, 51, 100 und 200. Dann gibt die Anfrage SucheUndNachfolger(10, 3, S) nacheinander die Elemente 10, 12, 50 und 51 aus.

Erweitern Sie AVL-Bäume, so dass SucheUndNachfolger(a, k, S) in Zeit $O(k + \log n)$ ausgeführt werden kann, wobei n die Anzahl der Knoten in S ist. Einfügen muss weiterhin in Zeit $O(\log n)$ möglich sein.

Beschreiben Sie die Idee Ihrer Erweiterung.

Geben Sie Algorithmen für Einfügen und SucheUndNachfolger an.

Hinweis: Kombinieren Sie AVL-Bäume mit doppelt verketteten Listen, um effizient auf den Nachfolger (im Sinne der Sortierung der Schlüssel) zugreifen zu können.

Aufgabe 3 (30 Punkte)**.**

1. Sei T ein B-Baum der Ordnung k und Höhe h. Bezeichne $\nu(T)$ die Anzahl der inneren Knoten von T. Zeigen Sie, dass für $h \geq 1$ gilt:
$$1 + 2 \cdot \frac{1 - k^{h-1}}{1-k} \leq \nu(T) \leq \frac{1}{2} \cdot \frac{(2k-1)^h - 1}{k-1}$$
2. Sei ein B-Baum der Ordnung k mit n Blättern durch $\mathcal{B}_k^n$ notiert. Bezeichne
$$\Delta_h(k,n) := \max\{|h - h'| : h = \text{Höhe}(\mathcal{B}_k^n), h' = \text{Höhe}(\widetilde{\mathcal{B}}_k^n)\}$$
den maximalen Höhenunterschied zweier B-Bäume der Ordnung k mit n Blättern.
Zeigen Sie, dass für $n \geq 3$ gilt:
$$\Delta_h(k,n) < \log_{2k-1}\left(\frac{n}{4}\right) + 1.$$

Aufgabe 4 (5 Punkte)**.** Konstruieren Sie einen erweiterten Trie für
$$S = \{a, an, ant, bar, baz, foo, fold, folding\}.$$

BITTE WENDEN!

Aufgabe 5 (10 Punkte)**.** Fügen Sie die Elemente $1, 10, 4, 24, 3, 100, 5, 30, 23, 15, 99$ in dieser Reihenfolge in eine Hashtabelle der Größe 11 ein.

1. Verwenden Sie dazu die Hashfunktion $h(n) := n \bmod 11$ und Kollisionsbehandlung mit verketteten Listen.
2. Verwenden Sie dazu offene Adressierung mit den Hashfunktionen

$$h(n, i) := (h_1(n) + ih_2(n)) \bmod 11, \quad h_1(n) := n \text{ und } \quad h_2(n) := n^2 \bmod 9 + 1.$$

Führen Sie alle Zwischenschritte auf!

Aufgabe 6 (25 Punkte)**.** Angenommen wir verwenden „Double Hashing“ zur Kollisionsbehandlung. Das heißt, wir verwenden eine Hashfunktion der Form $h(k, i) = (h_1(k) + ih_2(k)) \mod m$ für eine Hashtabelle mit m Einträgen. Zeigen Sie: Falls m und $h_2(k)$ den größten gemeinsamen Teiler $d \geq 1$ für einen festen Schlüssel k haben, dass die Hashfunktion $h(k, i)$ maximal $\frac{m}{d}$ Einträge der Hashtabelle besucht, bevor zu Eintrag $h(k, 0) = h_1(k)$ zurückgekehrt wird.

Bemerkung: Das bedeutet, falls $d = 1$, wird die gesamte Hashtabelle durchsucht.

$\square$