Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr/Dr. Sven Kosub
Moritz Maaß/Stefan Eckhardt

Wintersemester 2003/04
Übungsblatt 3
21. November 2003

Effiziente Algorithmen und Datenstrukturen I

Abgabetermin: 28.11.2003 **vor** *der Vorlesung*

Aufgabe 1

Sei T ein Binärbaum mit n Knoten. Wir definieren eine Nummerierung der Knoten $p : T \to \mathbb{N}$ mit

$$p(v) = \begin{cases} 1 & , v \text{ ist die Wurzel,} \\ 2p(u) & , v \text{ ist das linke Kind von } u \\ 2p(u)+1 & , v \text{ ist das rechte Kind von } u \end{cases}$$

Zeigen Sie, dass für alle Knoten eines echten Binärbaums T gilt:

$$p(v) \leq 2^{\frac{n+1}{2}} - 1$$

Aufgabe 2

Wir betrachten Stacks und Queues. Die Operationen Push, Pop, Enqueue, Dequeue können mit $O(1)$ Worst-Case Laufzeiten implementiert werden.

(a) Beschreiben Sie, wie eine Queue mit zwei Stacks implementiert werden kann, dass die Laufzeit für Enqueue und Dequeue amortisiert $O(1)$ ergibt.

(b) Beschreiben Sie, wie ein Stack mit zwei Queues implementiert werden kann. Welche Laufzeit ergibt sich für Push und Pop?

Aufgabe 3

Gegeben sei ein Baum mit n Knoten (implementiert als Linkstruktur). Der "lowest common ancestor" (LCA) zweier Knoten u und v ist definiert als der Knoten w mit der größten Tiefe, der sowohl Vorfahre von u als auch von v ist, d.h. kein Kind von w ist Vorfahre von u und von v.
Beschreiben Sie einen effizienten Algorithmus, der den LCA für zwei beliebige Knoten berechnet. Wie ist die Laufzeit Ihres Algorithmus?

Aufgabe 4

Sei T ein Baum mit n Knoten. Sei d_v die *Tiefe* des Knotens v in T, also der Abstand zur Wurzel. Der *Abstand* zweier Knoten u, v aus T ist $d(u, v) = d_v + d_u - 2d_w$, wobei w der LCA von u und v ist. Der *Durchmesser* von T ist definiert als der maximale Abstand zweier Knoten aus T: $diam(T) = \max_{u,v \in T} d(u, v)$. Beschreiben Sie einen effizienten Algorithmus zur Berechnung des Durchmessers eines Baums. Wie ist die Laufzeit Ihres Algorithmus?