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 7
19. Dezember 2003

Effiziente Algorithmen und Datenstrukturen I

Abgabetermin: 09.01.2004 **vor** *der Vorlesung*

Aufgabe 1

Beim normalen Splay-Tree gibt es zwei Phasen. In der ersten Phase wird der Knoten x, mit dem ein Splaying durchgeführt werden soll, in einer Suche von oben nach unten gefunden. In der zweiten Phase werden die Splay-Operationen von unten nach oben ausgeführt. Beschreiben Sie eine Methode, wie man die Such- und die Splay-Phase in einem einzigen Arbeitsgang von oben nach unten gleichzeitig durchführen kann. In jedem Schritt müssen dabei die nächsten zwei Knoten auf dem Pfad zu x untersucht werden, gefolgt von einem möglichen einzelnen zig-Schritt am Ende. Beschreiben Sie die Details für die verschiedenen Schritte zig-zig, zig-zag und zig.

Aufgabe 2

Angenommen wir haben eine sortierte Liste $S = (x_1, \ldots, x_n)$. Dabei sei den Elementen x_i ein Gewicht $a_i > 0$ in Form einer positiven, natürlichen Zahl zugeordnet. Sei $A = \sum_{i=1}^{n} a_i$. Geben Sie einen Algorithmus mit einer Laufzeit von $O(n \log n)$ an, der einen Suchbaum T für S konstruiert, so dass die Tiefe jedes Elements $O\left(\log\left(\frac{A}{a_i}\right)\right)$ ist.

Hinweis: Finden Sie das Element x_j mit maximalen j, so dass $\sum_{i=1}^{j-1} a_i < \frac{A}{2}$. Betrachten Sie, was passiert, wenn Sie dieses Element als Wurzel verwenden und dann rekursiv mit den beiden Teillisten weitermachen.

Aufgabe 3

Bei der Definition von Skip-Listen wurde mit der Wahrscheinlichkeit $\frac{1}{2}$ ein Element aus der Ebene i in die Ebene $i+1$ aufgenommen. Betrachten Sie denselben Algorithmus mit einer Wahrscheinlichkeit von p (mit $0 < p < 1$).

(i) Sei r die Anzahl von Listen. Beweisen Sie eine möglichst scharfe Schranke für r mit hoher Wahrscheinlichkeit (d.h. r soll mit Wahrscheinlichkeit $1 - o(1)$ beschränkt werden).

(ii) Geben Sie eine möglichst gute Schranke für die erwarteten Kosten pro Operation an.

(iii) Diskutieren Sie den Zusammenhang zwischen der Wahl von p und dem Laufzeitverhalten der Skip-Liste.

Aufgabe 4

Wir betrachten unsortierte Listen mit den Operationen $\mathsf{Insert}(x_i)$ („Füge x_i ein“), $\mathsf{Delete}(x_i)$ („Lösche x_i“) und $\mathsf{IsMember}(x_i)$ („Finde x_i“). Um auf ein Element zuzugreifen (Insert, IsMember), wird die Liste immer vom Anfang an durchsucht. Die Kosten sind also proportional zur Position des Elementes in der Liste.

Zeigen Sie, dass für eine gegebene Menge von Elementen $(x_1, \ldots, x_n)$ und eine feste Zugriffsfolge $\mathsf{IsMember}(x_{i_1}), \mathsf{IsMember}(x_{i_1}), \ldots, \mathsf{IsMember}(x_{i_m})$, die Liste, welche die Elemente in der Häufigkeit ihrer Zugriffe anordnet, optimal ist. (D.h. es gibt keine Listenordnung mit einer besseren Gesamtlaufzeit).