Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr
Matthias Baumgart

Wintersemester 2007/2008
Übungsblatt 6
23. November 2007

Effiziente Algorithmen und Datenstrukturen I

Abgabetermin: 30.11.2007 **vor** *der Vorlesung*

Aufgabe 1 (10 Punkte)

Implementieren Sie das zweistufige Hashverfahren von Fredman, Komlós und Szemerédi einschließlich einer IsElement-Funktion.

Das Universum der Schlüsselmenge sei $U = \{0, 1, \ldots, p-1\}$ mit $p = 2147483647$. Verwenden Sie die Klasse aller Hashfunktionen $h_{a,b} \colon U \to \{0, 1 \ldots, n-1\}$ mit

$$h_{a,b}(x) = ((a \cdot x + b) \bmod p) \bmod n \quad \text{für alle } x \in U,$$

wobei $0 < a < p$ und $0 \le b < p$.

Die zu speichernde Schlüsselmenge S sei in einer Datei enthalten, deren Name Sie beim Aufruf angeben. Die erste Zeile dieser Datei soll den Wert n, die zweite Zeile die Kardinalität der Schlüsselmenge S enthalten. Jede weitere Zeile enthält genau einen Schlüssel.

Erzeugen Sie Schlüsselmengen unterschiedlicher Größe und testen Sie Ihre Implementierung damit. Senden Sie anschließend den Quelltext sowie Ihre Testdaten per Email an `baumgart@in.tum.de`.

Aufgabe 2 (10 Punkte)

Geben Sie eine Folge von m Access-Operationen für eine Liste der Länge n an, welche die Transpose-Heuristik (TR) verwendet, an, so dass die amortisierten Kosten pro Access-Operation $\Omega(n)$ sind.

Aufgabe 3 (10 Punkte)

Wir modifizieren die Move-to-Front-Heuristik derart, dass ein zugegriffenes Element an Position k nicht an den Anfang der Liste, sondern nur den halben Weg nach vorn, genauer an Position $\lceil \frac{k+1}{2} \rceil$, gebracht wird (MHF-Heuristik). Zeigen Sie, dass

$$C_{MHF}(s) \le 2(2C_A(s) + X_A(s) - F_A(s) - m)$$

für jede Folge s von Zugriffsoperationen der Länge m gilt.

Aufgabe 4 (10 Punkte)

(a) Zeigen Sie, dass die Länge eines kürzesten Weges von der Wurzel zu einem Blatt in einem Leftist-Baum mit n Knoten höchstens $\lfloor \log(n+1) \rfloor$ ist.

(b) Beschreiben Sie einen Algorithmus, der zwei Leftist-Bäume T_1 und T_2, in denen jeweils die Heapbedingung gilt und deren Schlüsselmengen disjunkt sind, zu einem Leftist-Baum T vereinigt. Analysieren Sie die Laufzeit dieser Merge-Operation.

Hinweis: Nehmen Sie an, dass Sie für jeden Knoten x eines Leftist-Baumes die Länge eines kürzesten Weges zu einem Blatt kennen.