Prof. Dr. A. Goerdt
L. Falke

TU Chemnitz
Sommersemester 2015
02.04.2015

Effiziente Algorithmen / Theoretische Informatik III

3. Übung

1. Aufgabe: Geben Sie eine Familie von Operationsfolgen an, bei denen die Kosten der Heuristiken `B`, `TR` und `FC` nicht O(Kosten der Heuristik `MF`) sind.

2. Aufgabe: Was ist die Worst-Case-Laufzeit der Heuristik `MF` bei m Operationen und anfangs leerer Liste?

3. Aufgabe: Führen Sie den Beweis des folgenden Satzes für $\sigma_i =$`Delete`(x) aus:

> Sei σ eine beliebige Folge von Dictionary-Operationen $\sigma_1, \cdots, \sigma_i, \cdots, \sigma_n$ auf einer anfangs leeren Liste. Die Kosten des Algorithmus A für σ seien mit $K_{\mathcal{A}}(\sigma)$ und die für `Move-to-Front` seien mit $K_{\mathcal{MF}}(\sigma)$ bezeichnet.
>
> Dann gilt für alle Algorithmen A
>
> $$K_{\mathcal{MF}}(\sigma) \leq 2 \cdot K_{\mathcal{A}}(\sigma).$$

4. Aufgabe: Wir untersuchen, wie sich der Splaybaum auf dem Anfangsbeispiel der Vorlesung verhält:
Fügen Sie die Elemente $n, n-1, n-2, ..., 1$ in einen anfänglich leeren Splaybaum ein. Führen Sie danach `Find`(n) aus. Überlegen Sie sich, daß die Kosten der zugehörigen Folge von Operationen tatsächlich

$$O((n+1) \cdot \log\ n)$$

betragen. Überlegen Sie sich, wie die darauffolgenden `Find`$(n-1)$, ...,`Find`(1) bearbeitet werden. Wie kann danach wieder ein „ganz dünner“ Baum entstehen?