11. Aufgabenblatt zur Vorlesung

Höhere Algorithmik WiSe 2011/12

Wolfgang Mulzer, Claudia Dieckmann, Romain Grunert, Tillmann Miltzow, Lena Schlipf

Abgabe am 20. Januar 2012 vor der Vorlesung in die Mitarbeiterschachteln

Aufgabe 1 Quake Heaps: Details *10 Punkte*

Beschreiben Sie die Details von Quake Heaps. Gehen Sie dabei auf die folgenden Fragen ein: Wie genau werden die Einträge im Heap gespeichert? Was wird in jedem Knoten des Turnierbaumes gespeichert? Welche Schritte werden genau bei `insert`, `delete-min` und `decrease-key` ausgeführt? Wann und wie müssen die Zeiger im Heap aktualisiert werden? Wann und wie werden die Arrays `T[i]` und `n[i]` aktualisiert?

Aufgabe 2 Quake Heaps: Analyse *10 Punkte*

Der Schritt in `delete-min`, der dafür sorgt, dass es zu jeder Höhe höchstens einen Baum gibt, wird *Konsolidierung* genannt. In dieser Aufgabe sollen Sie sich über verschiedene Varianten der Konsolidierung Gedanken machen.

(a) Nehmen Sie an, wir überspringen in `delete-min` den Konsolidierungsschritt. Wie ändert sich dadurch die Laufzeitanalyse?

(b) Nehmen Sie an, dass wir am Ende von `delete-min` eine weitere Konsolidierung durchführen, falls ein Erdbeben stattgefunden hat. Zeigen Sie, dass sich dadurch die amortisierten Laufzeiten asymptotisch nicht ändern.

(c) (*freiwillig, 5 Zusatzpunkte*) Was passiert, wenn man die Konsolidierung statt in `delete-min` am Ende von `insert` durchführt?

Aufgabe 3 Potentialfunktionen *10 Punkte*

In der Vorlesung haben wir Quake Heaps mit Hilfe der Buchhaltermethode analysiert. Eine alternative Technik verwendet so genannte *Potentialfunktionen.* Wir betrachten eine Datenstruktur D. Sei $\mathcal{D}$ die Menge aller möglichen Zustände von D. Eine Potentialfunktion $\Phi : \mathcal{D} \to \mathbb{N}_0$ ist eine Funktion, die jedem Zustand $D \in \mathcal{D}$ der Datenstruktur eine natürliche Zahl $\Phi(D)$ zuordnet. Für den ursprünglichen Zustand D_0 muss gelten: $\Phi(D_0) = 0$.

Wir definieren nun die *amortisierten Kosten* einer Operation X auf der Datenstruktur. Sei D die Datenstruktur vor der Operation X und D' die Datenstruktur nach X. Seien c_X die tatsächlichen Kosten von X. Dann sind die amortisierten Kosten von X, $\hat{c}_X$, definiert als $\hat{c}_X := c_X + \Phi(D') - \Phi(D)$.

(a) Geben Sie eine Interpretation der Potentialfunktion und erklären Sie die Intuition hinter der obigen Definition der amortisierten Kosten.

(b) Sei X_1, X_2,, …, X_n eine Folge von Operationen, die auf der initialen Datenstruktur D_0 ausgeführt wird. Zeigen Sie, dass $\sum_i c_{X_i} \leq \sum_i \hat{c}_{X_i}$ ist. Interpretieren Sie das Ergebnis.

(c) Betrachten Sie das Binärzählwerk aus Aufgabe 2(a) auf dem 8. Aufgabenblatt. Sei Φ die Funktion, die einem Zustand des Binärzählwerks die Anzahl der Einsen im aktuellen Zählerstand zuordnet.

Verwenden Sie Φ, um zu zeigen, dass die amortisierten Kosten pro Zählvorgang konstant sind. Folgern Sie, dass das Binärzählwerk mit Kosten $O(n)$ von 0 bis n zählen kann. Vergleichen Sie Ihren Beweis hier mit Ihrem Beweis vom 8. Aufgabenblatt.

Aufgabe 4 Quake Heaps: Implementierung *10 Zusatzpunkte*

Implementieren Sie Quake Heaps in Java. Führen Sie geeignete Experimente durch und dokumentieren Sie diese.