9. Aufgabenblatt zur Vorlesung

Algorithmen, Datenstrukturen und Datenabstraktion WS 17/18

Wolfgang Mulzer, Katharina Klost

Abgabe am 22. Dezember 2017 bis 10 Uhr in die jeweiligen Tutorenfächer

Aufgabe 1 AVL-Bäume *10 Punkte*

(a) Fügen Sie die Schlüssel A, L, P, D, R, E, I, F, O, M, N in dieser Reihenfolge in einen anfangs leeren AVL-Baum ein. Löschen Sie sodann die Schlüssel A, M, R. Zeichnen Sie den Baum nach jedem Einfüge- und Löschvorgang, und zeigen Sie die Rotationen, welche durchgeführt werden. Annotieren Sie dabei auch die Knoten mit ihrer jeweiligen Höhe.

(b) Beweisen Sie: Beim Einfügen in einen AVL-Baum wird höchstens eine (Einfach- oder Doppel-)Rotation ausgeführt. Gilt das auch beim Löschen (Begründung)?

Aufgabe 2 Gewichtsbalancierte Bäume *10 Punkte*

Sei T ein binärer Suchbaum mit n Knoten. Wir definieren eine *Gewichtsfunktion* auf T. Es ist $w(\perp) = 1$. Sei v ein Knoten von T, mit linkem Kind v_l und rechtem Kind v_r (es ist $v_l = \perp$ oder $v_r = \perp$, falls das jeweilige Kind nicht vorhanden ist). Dann ist $w(v) = 1 + w(v_l) + w(v_r)$.

Wir nennen T *gewichtsbalanciert*, falls für alle Knoten v in T gilt: Seien v_l und v_r das linke und das rechte Kind von v. Dann ist $w(v_l) \geq w(v)/10$ und $w(v_r) \geq w(v)/10$.

Zeigen Sie: Wenn T gewichtsbalanciert ist, dann hat T Höhe $O(\log n)$.

Hinweis: Verwenden Sie Induktion. Argumentieren Sie zunächst, dass $w(v_l) \leq \frac{9}{10}w(v)$ und $w(v_r) \leq \frac{9}{10}w(v)$ sind.

Aufgabe 3 Treaps/Balden/Baufen *10 Punkte*

Ein Treap/eine Balde/ein Baufen ist eine Kreuzung aus einem **Tr**ee (bzw. **B**aum) und einem H**eap** (bzw. einer H**alde**/einem H**aufen**). Die Knoten eines Treaps T speichern (neben den Werten) zwei Felder: einen *Schlüssel* und eine *Priorität*. Bezüglich der Schlüssel hat T die binäre Suchbaumeigenschaft, bezüglich der Prioritäten die Min-Heap-Eigenschaft. Das heißt, für jeden inneren Knoten v von T gilt: (i) alle Knoten im linken Unterbaum von v haben Schlüssel kleiner als der Schlüssel von v; (ii) alle Knoten im rechten Unterbaum von v haben Schlüssel größer als der Schlüssel von v; (iii) die Prioritäten der Kinder von v sind größer als die Priorität von v.

(a) Zeichnen Sie einen Treap für die folgende Menge von Schlüssel-Prioritäts-Paaren: (E, 23), (K, 65), (H, 5), (B, 7), (I, 73), (A, 10), (G, 4).

(b) Zeigen Sie: Wenn in einer Menge von Schlüssel-Prioritäts-Paaren alle Schlüssel und Prioritäten paarweise verschieden sind, so gibt es genau einen Treap für diese Menge.

(c) Geben Sie Pseudocode für Algorithmen zum Einfügen und Löschen in einen Treap.

    Hinweis: Führen Sie zunächst die Operation genauso durch wie in einem herkömmlichen binären Suchbaum. Verwenden Sie dann Einfachrotationen, um die Heapeigenschaft wieder herzustellen. Pseudocode für binäre Suchbäume finden Sie auf der Veranstaltungsseite.

Bemerkung: Man kann zeigen, dass Treaps erwartete Höhe $O(\log n)$ haben, wenn die Prioritäten zufällig aus dem Intervall $(0, 1)$ gewählt werden.

Hinweis: Bitte formatieren Sie Ihre Abgaben mit LaTeX oder einem vergleichbaren elektronischen Textverarbeitungssystem. Ansonsten droht der Abzug von bis zu 10% der erreichbaren Punkte.