technische universität dortmund

SoSe 2018

M. Buchin
A. Krivošija, D. Mezlaf
L. Pfahler, A. Rey

DAP2 – Heimübung 10

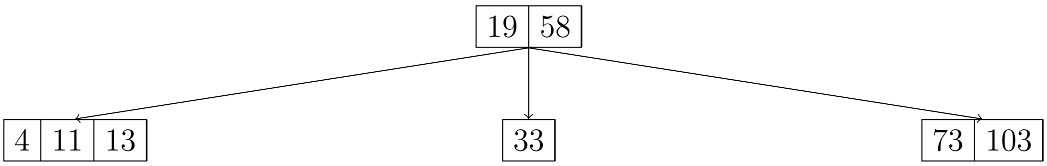
Ausgabedatum: 08.06.2018 — Abgabedatum: Mo. 18.06.2018 bis 12 Uhr

Abgabe:

Schreiben Sie unbedingt immer Ihren vollständigen Namen, Ihre Matrikelnummer und Ihre Gruppennummer auf Ihre Abgaben! Beweise sind nur dort notwendig, wo explizit danach gefragt wird. Eine Begründung der Antwort wird allerdings *immer* verlangt.

Aufgabe 10.1 (5 Punkte): (B-Bäume)

Gegeben sei der folgende B-Baum T mit Parameter $t = 2$:

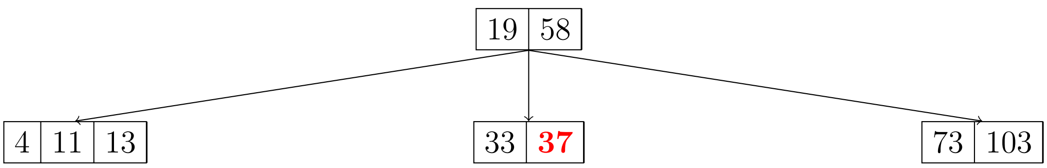


Fügen Sie in der angegebenen Reihenfolge die Schlüssel 37, 42, 7 und 17 ein. Geben Sie für jede Operation die ggf. nötigen Split-Operationen sowie den resultierenden B-Baum an.

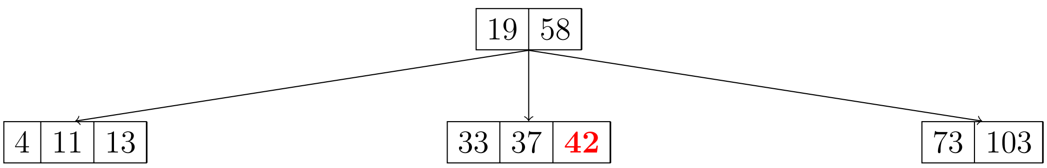
Lösung:

Bei jedem Einfügen führen wir zunächst die Suche nach dem Element durch, um herauszufinden, ob ein Split nötig ist.

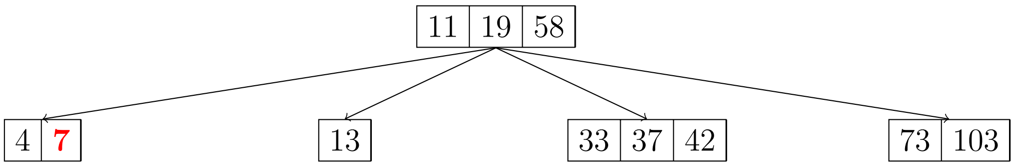
Füge 37 ein: Auf dem Suchpfad zur 37 (Wurzel, mittleres Kind) liegt kein Knoten mit der maximalen Schlüsselzahl von $2t - 1 = 3$ Schlüsseln, es muss also kein Split durchgeführt werden. Wir fügen 37 in das mittlere Blatt ein:



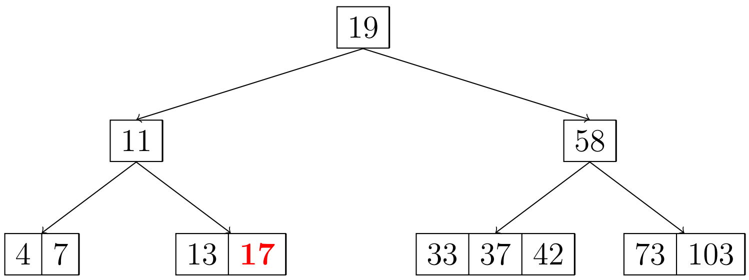
Füge 42 ein: Auf dem Suchpfad zur 42 (Wurzel, mittleres Kind) liegt kein Knoten mit der maximalen Schlüsselzahl von $2t - 1 = 3$ Schlüsseln, es muss also kein Split durchgeführt werden. Wir fügen 42 in das mittlere Blatt ein:



Füge 7 ein: Auf dem Suchpfad zur 7 liegt die Wurzel mit zwei und das linke Blatt mit drei Schlüsseln, das Blatt muss also gesplittet werden. Der Median der drei Schlüssel ist 11, dieser Schlüssel wird in die Wurzel geschoben, die beiden weiteren Teilknoten werden zu Blättern. Anschließend wird der Schlüssel 7 zum Blatt mit dem Schlüssel 4 hinzugefügt:



Füge 37 ein: Auf dem Suchpfad zum Schlüssel 17 finden wir diesmal die Wurzel mit drei Schlüsseln, diese muss also gesplittet werden. Anschließend wird die 17 zum Blatt mit dem Schlüssel 13 hinzugefügt:



Aufgabe 10.2 (5 Punkte): (Einfügen in balancierten Bäumen)

Beantworten Sie folgende Fragen, indem Sie jeweils eine möglichst kleine obere Schranke angeben, und beweisen Sie jeweils Ihre Aussage.

a) Wie viele Split-Operationen kann es beim Einfügen eines Schlüssels in einen B-Baum vom Grad t mit bisher n Schlüsseln maximal geben?

b) Wie viele Rotationen werden maximal beim Einfügen eines Schlüssels in einen AVL-Baum mit bisher n Schlüsseln benötigt?

Lösung:

a) *Behauptung:* Beim Einfügen eines Schlüssels in einen B-Baum vom Grad t mit bisher n Schlüsseln können maximal $\log_t \left({}^{(t-1)n}/_{(2t-1)} + 1\right)$ Split-Operationen vorkommen.

Beweis: Wir schätzen zunächst die Anzahl der Knoten im B-Baum in Abhängigkeit von Grad t, der Höhe h und der Anzahl s an Split-Operationen nach unten ab und schließen daraufhin auf eine obere Schranke für s abängig von n und t.

Eine Split-Operation muss durchgeführt werden, wenn ein Knoten auf dem Suchpfad voll ist, also $2t-1$ Schlüssel enthält. Auf dem Suchpfad liegen also mindestens $s \cdot (2t-1)$ Schlüssel. Jeder Knoten, der aufgeteilt wird, befindet sich jeweils auf einer anderen Ebene g, $0 \leq g \leq h$, und hat außerdem $2t-1$ Kinder, die nicht auf dem Suchpfad liegen. Diese sind jeweils Wurzelknoten von Teil-B-Bäumen der Höhe $g-1$ (leerer Baum für $g = 0$). Die minimale Anzahl m_g der Knoten in einem Baum der Höhe g ist laut Vorlesung durch $2t^g - 1$ nach unten beschränkt. Da hier die Wurzel aber ein innerer Knoten ist, enthält ein solcher Teilbaum mindestens $t^{g+1} - 1$ Schlüssel. Das gilt, weil ein inneres Knoten eines B-Baums mindestens $t - 1$ Schlüssel enthalten muss. Analog zur Herleitung der Formel aus der Vorlesung hat die Wurzel dieses Teilbaums mindestens t Kinderknoten, was $(t-1) \cdot t$ Knoten auf der ersten Unterebene entspricht, $(t-1) \cdot t^2$ Knoten auf der zweiten Unterebene, usw. bis $(t-1) \cdot t^g$ in Blätterknoten (der Teilbaum hat die Höhe g). Insgesamt macht das

$$(t-1) \cdot \sum_{i=0}^{g} t^i = (t-1) \cdot \frac{t^{g+1} - 1}{t - 1} = t^{g+1} - 1$$

Schlüssel in solchem Teilbaum. Die gesamte Anzahl n aller Knoten ist minimal, wenn die Höhen g minimal sind, also sich auf dem Suchpfad nur zu splittende Knoten befinden, d. h., $s = 1 + h$. Insgesamt ergibt sich also unter diesen Bedingungen folgende untere Schranke für die Knotenanzahl n:

$$\begin{aligned}
n \geq & (1+h)(2t-1) + \sum_{g=0}^{h-1}(2t-1)m_g = (1+h)(2t-1) + \sum_{g=0}^{h-1}(2t-1)(t^{g+1}-1) \\
= & (1+h)(2t-1) + t(2t-1)\left(\sum_{g=0}^{h-1} t^g\right) - (2t-1)\sum_{g=0}^{h-1} 1 \\
= & (1+h-h)(2t-1) + t(2t-1)\frac{t^h - 1}{t-1} = (2t-1)\frac{t-1}{t-1} + (2t-1)\frac{t^{h+1} - t}{t-1} \\
= & \frac{2t-1}{t-1}\left(t^{h+1} - 1\right).
\end{aligned}$$

Daraus ergibt sich

$$t^{h+1} \leq \frac{(t-1)n}{2t-1} + 1.$$

Also

$$s \leq \log_t \left(\frac{(t-1)n}{(2t-1)} + 1\right).$$

Bemerkung: Diese Schranke kann genau erreicht werden, da es einen B-Baum mit vollen Knoten auf einem Suchpfad geben kann, aber Knoten mit minimaler Schlüsselanzahl sonst.

b) *Behauptung:* Beim Einfügen eines Schlüssels in einen AVL-Baum mit bisher n Schlüsseln werden maximal 2 Rotationen benötigt.

Aus der Vorlesung ist bekannt, dass beim Einfügen an jedem Knoten auf dem Suchpfad (von Blatt zur Wurzel) balanciert wird, falls der Unterbaum ein Beinahe-AVL-Baum ist. Wir zeigen nun, dass dies höchstens einmal nötig ist.

Sei T ein AVL-Baum, und sei v ein Knoten, der eingefügt wird und damit zu einer Verletzung der AVL-Eigenschaft in mindestens einem Knoten von T führt. Sei x der tiefste Knoten auf dem Suchpfad zu v, der nicht in Balance ist. Dabei bezeichnen wir mit der Tiefe eines Knotens x in T seinen Abstand zur Wurzel von T. Dann ist die Differenz in x genau 2, und der Teilbaum von x, in dem sich v befindet, ist der höhere. Sei d die Tiefe von v in T. Dann ist v der einzige Knoten mit Tiefe d unter dem Knoten x, da er die Höhe eines Teilbaums unter x geändert haben muss, um den Knoten x aus der Balance zu bringen. Sei nun v o.B.d.A. im linken Teilbaum von x. (Ist v im rechten Teilbaum, läuft die folgende Argumentation symmetrisch.) Das linke Kind von x ist per Definition des Knotens x balanciert. Je nachdem, ob v sich nun im Baum unter $lc(lc(x))$ oder $rc(lc(x))$ befindet, wird eine einfache oder eine Doppelrotation durchgeführt. Folgende Aussagen werden durch Durchspielen der beiden möglichen Vorgehensweisen der Balance-Operation (Einzel- oder Doppelrotation) auf x gezeigt:

- Knoten x ist anschließend balanciert.
- Die Höhe des Teilbaumes mit Wurzel x wird durch die Rotationen um 1 verringert.

Die zweite Aussage liefert, dass jeder Knoten über x, der durch das Einfügen von v aus der Balance gebracht wurde, anschließend wieder in Balance ist, da unter jedem solchen Knoten der Teilbaum mit v der schwerere gewesen sein muss. Insgesamt ist also nach diesen zwei Rotationen jeder Knoten im AVL-Baum T balanciert, sodass die Einfüge-Operation keine weiteren Rotationen durchführt.

Pro Balance-Operation sind maximal zwei Rotationen nötig, also insgesamt maximal zwei Rotationen.