Datenstrukturen und Algorithmen SS14
Übungsblatt 6 (Abgabe 02.06.2014)

Prof. Dr. E. Ábrahám
F. Corzilius, S. Schupp, T. Ströder

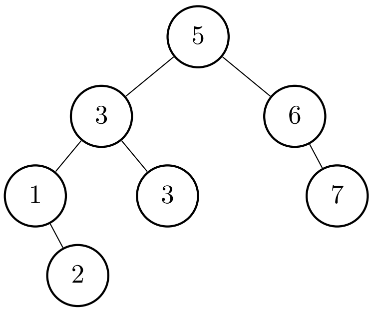
Allgemeine Hinweise:

- Die **Hausaufgaben** sollen in Gruppen von je **2 bis 3 Studierenden** aus der **gleichen Kleingruppenübung (Tutorium)** bearbeitet werden. **Namen und Matrikelnummern** der Studierenden sind auf jedes Blatt der Abgabe zu schreiben. **Heften bzw. tackern Sie die Blätter!**
- Die **Nummer der Übungsgruppe** muss **links oben** auf das **erste Blatt** der Abgabe geschrieben werden. Notieren Sie die Gruppennummer gut sichtbar, damit wir besser sortieren können.
- Die Lösungen müssen **bis Montag, den 02.06.2014 um 9:00 Uhr** in den entsprechenden Übungskasten eingeworfen werden. Sie finden die Kästen am Eingang Halifaxstr. des Informatikzentrums (Ahornstr. 55). Alternativ können Sie die Lösungen auch in Ihrem Tutorium vor der Abgabefrist direkt bei Ihrer Tutorin/Ihrem Tutor abgeben.

Tutoraufgabe 1 (AVL–Bäume):

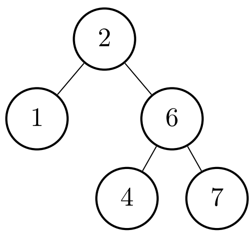
a) Fügen Sie die Schlüssel 1, 3, 5, 6, 6, 4 und 2 in dieser Reihenfolge in einen initial leeren AVL Baum ein. Geben Sie den Baum direkt nach dem Einfügen sowie nach jeder Rotation an.

b) Löschen Sie den Knoten, den die Suche nach dem Schlüssel 3 ausgibt, aus dem folgenden AVL Baum. Geben Sie den Baum direkt nach dem Löschen sowie nach jeder Rotation an.

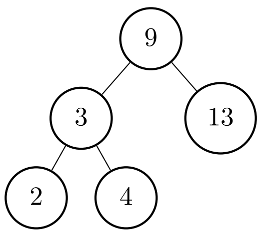


Aufgabe 2 (AVL–Bäume): (1,5 + 1,5 + 1,5 + 1,5 = 6 Punkte)

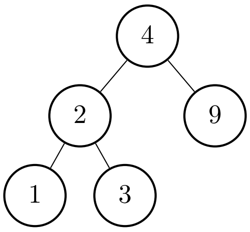
a) Fügen Sie den Schlüssel 5 in den folgenden AVL Baum ein. Geben Sie den Baum direkt nach dem Einfügen sowie nach jeder Rotation an.



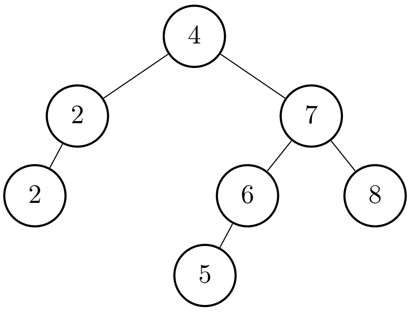
b) Fügen Sie den Schlüssel 3 in den folgenden AVL Baum ein. Geben Sie den Baum direkt nach dem Einfügen sowie nach jeder Rotation an.



c) Löschen Sie den Knoten, den die Suche nach dem Schlüssel 4 ausgibt, aus dem folgenden AVL Baum. Geben Sie den Baum direkt nach dem Löschen sowie nach jeder Rotation an.



d) Löschen Sie den Knoten, den die Suche nach dem Schlüssel 2 ausgibt, aus dem folgenden AVL Baum. Geben Sie den Baum direkt nach dem Löschen sowie nach jeder Rotation an.

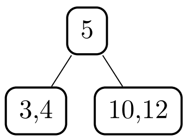


Tutoraufgabe 3 (2–3–4–Bäume):

Fügen Sie in der angegebenen Reihenfolge die Schlüssel $8, 3, 15, 5, 12, 14, 10$ in einen anfangs leeren 2–3–4–Baum ein.

Aufgabe 4 (2–3–4–Bäume): (3 + 1 + 1 = 5 Punkte)

a) Fügen Sie in der angegebenen Reihenfolge die Schlüssel $9, 2, 14, 6, 11, 13, 0, 8, 7, 1$ in den folgenden 2–3–4–Baum ein und geben Sie die dadurch jeweils entstehenden 2–3–4–Bäume an:



b) Geben Sie für die Zahlen 1 bis 9 eine Einfügereihenfolge in einen anfangs leeren 2–3–4–Baum an, sodass dieser seine maximale Höhe erreicht. Welche Höhe ist dies?

c) Geben Sie für die Zahlen 1 bis 9 eine Einfügereihenfolge in einen anfangs leeren 2–3–4–Baum an, sodass dieser eine minimale Anzahl an Knoten aufweist. Wie viele Knoten sind dies?

Tutoraufgabe 5 (Knobeleien):

a) Wir haben einen binären Suchbaum, in dem die Zahlen von 1 bis 1000 einsortiert sind. Wir suchen nach der Zahl 412. Welche der folgenden Sequenzen von Schlüsselwerten kann **nicht** eine überprüfte Knotenfolge sein?

i 998, 14, 36, 512, 247, 309, 499, 412

ii 565, 501, 181, 673, 500, 480, 427, 412

iii 2, 837, 547, 137, 230, 453, 402, 412

iv 765, 699, 643, 555, 270, 315, 411, 412

v 666, 245, 598, 301, 572, 365, 500, 412

vi 666, 182, 575, 194, 483, 317, 595, 412

vii 2, 781, 776, 633, 215, 545, 214, 412

b) Gegeben sei ein beliebiger binärer Suchbaum bt. Angenommen die Suche nach einem Element endet in einem Blatt r. Sei nun A die Menge der Schlüssel, die links vom Suchpfad liegen, B die Menge der Schlüssel, die auf dem Suchpfad liegen und C die Menge der Schlüssel, die rechts vom Suchpfad liegen.

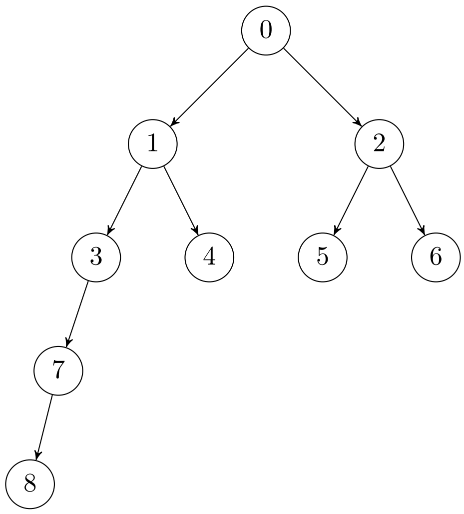
Behauptung: Jedes Tripel $a \in A, b \in B, c \in C$ erfüllt $a \leq b \leq c$.

Finden Sie ein möglichst kurzes Gegenbeispiel zu der Behauptung.

Aufgabe 6 (Knobeleien): (3 + 5 + 7* = 8 + 7* Punkte)

Ein Binärbaum ist c-höhenbalanciert, wenn der Unterschied in der Tiefe des rechten und des linken Teilbaumes jedes Knotens c nicht überschreitet.

Beispiel (2-höhenbalancierter Binärbaum):



a) Geben Sie einen 2-höhenbalancierten Binärbaum der Tiefe 5 mit der minimalen Anzahl an Blättern an.

b) Beschreiben Sie eine (rekursive) Methode, um beliebige c-höhenbalancierte Binärbäume der Tiefe k mit minimaler Anzahl an Blättern zu erstellen.

c)* Zeigen Sie, dass Ihre Methode c-höhenbalancierte Binärbäume mit der minimalen Anzahl an Blättern erzeugt.