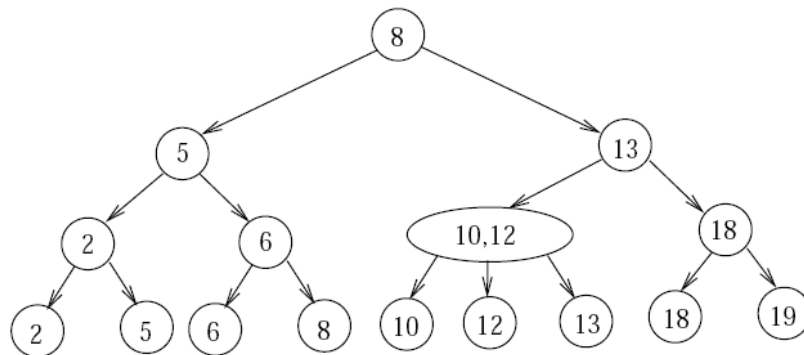


Übungen zur Vorlesung
Datenstrukturen, Algorithmen und Programmierung 2 (DAP2)
 Sommersemester 2007

Blatt 6

Aufgabe 6.1 (5 Punkte)

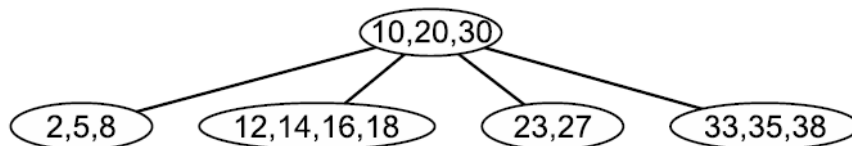
Betrachte folgenden 2-3-Baum in der im Skript beschriebenen Variante, bei der alle Daten an den Blättern stehen und an den inneren Knoten das größte Datum des linken Teilbaums und, sofern vorhanden, auch das größte Datum des mittleren Teilbaums als Routinginformation stehen:



Führe die Operation SPLIT(12) aus und dokumentiere alle wesentlichen Zwischenschritte.

Aufgabe 6.2 (5 Punkte)

a) Gegeben sei der folgende B-Baum der Ordnung 5:



Bestimme den durch die gegebene Folge von Operationen entstehenden B-Baum; gib auch die Bäume nach den einzelnen Schritten an.

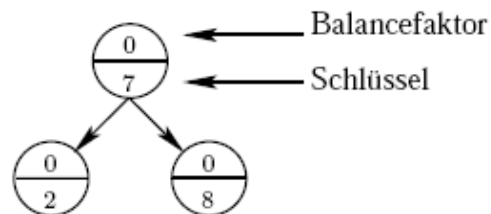
INSERT(37), INSERT(31), INSERT(15), DELETE(2), DELETE(15)

- b) Ein Studienkollege erzählt Dir von seiner Idee, das balancierende Schema der AVL-Bäume auf B-Bäume zu erweitern, also ebenfalls Baumknoten zu rotieren. Dadurch würde man die selben Vorteile erhalten, wie beim Schritt von binären Suchbäumen zu AVL-Bäumen. Dieser Student fiel schon immer durch Ideen der selben Güte auf. Sollte dieser Student nächstes Jahr eher DAP2-HiWi werden oder DAP2 wiederholen? Begründe Deine Antwort.

Aufgabe 6.3 (5 Punkte)

Gegeben sei der in der Abbildung dargestellte AVL-Baum. Füge die Schlüssel [11, 22, 9, 0, 4, 5, 3] in der angegebenen Reihenfolge in den Baum ein. Zeichne pro Einfügeoperation den Baum, gib die notwendige Rotation (einfach oder doppelt) an und veranschauliche die Rotation in dem Baum. Notiere dabei in jedem Knoten den Balancefaktor.

Hinweis: Achte darauf, das Rebalancieren mit dem richtigen Knoten zu beginnen.



Aufgabe 6.4 (5 Punkte)

- a) In einem inneren Knoten eines B-Baums der Ordnung M werden bekanntlich immer mehrere Schlüssel gespeichert. Beim Aufruf der SEARCH-Funktion müssen diese analysiert werden, um zu wissen, in welchem Teilbaum die Suche fortgesetzt werden muss. Dazu betrachte die beiden Ansätze der linearen und der binären Suche. Wie beeinflusst deren Wahl die Ausführung von n SEARCH-Funktionen (mit jeweils zufällig ausgewähltem Suchparameter) in O -Notation ?
- b) Nehmen wir an, dass wir die Ordnung eines B-Baums während der Einfügeoperation erhöhen dürfen. Dadurch können wir sicherstellen, dass die Ordnung immer mindestens halb so groß ist, wie die Anzahl der Elemente im Baum. Offensichtlich können wir dadurch die Tiefe im Baum konstant halten. Da die Suche im Baum linear in der Tiefe des Baums ist, können wir dadurch in konstanter Zeit suchen. -Falls du damit nicht einverstanden bist, erkläre den Fehler.
- c) Wir definieren einen „Bruder“ des B-Baums, den $B_{2/3}$ -Baum der Ordnung $M \geq 6$. Dieser unterscheidet sich von ersterem dadurch, dass jeder Knoten (ausser der Wurzel) mindestens $\lceil \frac{2}{3}M \rceil + 1$ Kinder hat. Beschreibe die Vor- und Nachteile dieses Baumtyps.