

## Datenstrukturen und Algorithmen – Informatik I

### 9. Übung

Abgabe der Lösungen: 5. Juli 2004

---

**Hinweis:** Die Implementierungsaufgaben sind in Java oder C/C++ zu lösen. Der Quellcode sowie ein dokumentierter Beispiellauf sind beim Betreuer der Kleingruppe abzugeben.

#### Aufgabe 39: Doppeltes Hashing

3 Punkte

Gegeben sei die Hashfunktion

$$h(k) = k \bmod 11.$$

Fügen Sie die Schlüsselfolge

16, 44, 21, 5, 19, 22, 8, 33, 27, 30

gemäß der Hashfunktion  $h(k)$  in eine geschlossene Hashtabelle ein. Verwenden Sie Doppel-Hashing als Kollisionsstrategie, indem Sie eine geeignete Hashfunktion  $h'(k)$  wählen. Stellen Sie das Ergebnis graphisch dar.

#### Aufgabe 40: Programmieraufgabe: Binäre Suchbäume

4 Punkte

Geben Sie einen Algorithmus an, der die  $k$  kleinsten Elemente eines binären Suchbaumes ausgibt. Dabei soll die Laufzeit des Algorithmus zum Finden des  $k$ -kleinsten Elements  $O(h)$  betragen ( $h$  ist die Höhe des binären Suchbaumes). Implementieren Sie den Algorithmus und testen Sie Ihre Implementierung an den folgenden Beispieldaten:

18	-5	-8	-3	121	-30	100	-89	20	-100	10	105	-34	2
-105	200	300	-611	500	-222	-299	-495	-125	501	-798	-30	450	13

Für diese Aufgabe können Sie die Klasse *BinaryTree* und insbesondere die Methode *Insert* aus der Aufgabe 9, Übung 2, modifizieren und verwenden.

*Tip:* Einen Hinweis zu dem Algorithmus finden Sie im Vorlesungsskript auf Seite 133 (Operation  $Order(k, S)$ ).

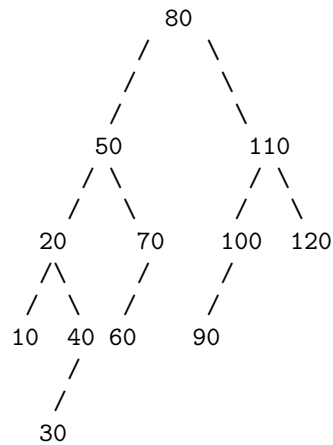
#### Aufgabe 41: Optimaler Suchbaum

4 Punkte

Erstellen Sie (per Hand) die Tabelle, die sich bei der Berechnung des optimalen Suchbaumes mittels dynamischer Programmierung für  $S = \{1, 2, 3, 4, 5\}$  mit  $\alpha_0 = 4, \beta_1 = 1, \alpha_1 = 0, \beta_2 = 3, \alpha_2 = 2, \beta_3 = 8, \alpha_3 = 5, \beta_4 = 10, \alpha_4 = 0, \beta_5 = 3, \alpha_5 = 3$  ergibt. Ermitteln Sie hieraus den optimalen Suchbaum und stellen Sie diesen graphisch dar.

**Aufgabe 42: Binäre Suchbäume****2+4 Punkte**

Gegeben sei der folgende binäre Suchbaum:



Zeigen Sie die Änderungen, die sich an diesem Baum ergeben, wenn zunächst die Zahlen

35, 85, 105, 5, 15, 45, 130, 140

eingefügt und anschließend die Zahlen

35, 20, 10

gelöscht werden. Behandeln Sie den Baum dabei

- als normalen binären Suchbaum
- als AVL-Baum und geben Sie an, welche Rotationen durchgeführt werden.

**Aufgabe 43: AVL-Bäume****1+3+2 Punkte**

- Zeichnen Sie für jede Baumhöhe  $h = 1, 2, \dots, 7$  einen AVL-Baum mit minimaler Zahl der Knoten (d.h. einen maximal entarteten AVL-Baum dieser Höhe) auf.
- Zeichnen Sie außerdem *alle* AVL-Bäume der Höhe 3 mit minimaler Zahl von Knoten.
- Geben Sie allgemein für die Baumhöhe  $h$  die Anzahl der verschiedenen AVL-Bäume mit minimaler Zahl von Knoten an (ohne Beweis).