



# Informatik II: Algorithmen und Datenstrukturen

SS 2002

## 9. Übungsblatt

### Aufgabe 26 (AVL-Bäume, 4T+3P)

- Konstruieren Sie ein Beispiel, bei dem das Löschen eines Elementes aus einem AVL-Baum zwei (einfache oder doppelte) Rotationen erfordert. Skizzieren Sie den Baum vorher, nach der ersten Rotation und nachher.
- Betrachten Sie die in der Vorlesung vorgestellten Operationen für AVL-Bäume. Ergänzen Sie eine Funktion

```
int avl_hoehe (avl_baum b)
```

die mit möglichst geringem Aufwand die Höhe eines AVL-Baums bestimmt. Wenden Sie die Funktion auf folgendes Beispiel an, indem Sie nach jeder Operation die Höhe des entstandenen AVL-Baums bestimmen:

- leeren Baum erzeugen
- nacheinander Datensätze mit Schlüsseln 1, 2, ..., 9 einfügen
- nacheinander die Datensätze mit den Schlüsseln 2, 9 und 1 löschen.

Analysieren Sie den Aufwand der Funktion `avl_hoehe` abhängig von  $n$  ( $\mathcal{O}$ -Analyse genügt).

### Aufgabe 27 (Hashing, 6T)

- Entwerfen Sie eine Datenstruktur mit folgenden Eigenschaften: Wenn bereits  $n$  Datensätze gespeichert sind, erfordert das Suchen, Einfügen bzw. Löschen eines Datensatzes

$$T_{WC} = \mathcal{O}(\log n) \text{ Schlüsselvergleiche bzw. Operationen.}$$

Im Mittel sind sogar nur

$$T_{AC} = \mathcal{O}(1) \text{ Schlüsselvergleiche bzw. Operationen}$$

notwendig, sofern  $n = \mathcal{O}(\text{h\_size})$ , `h_size` eine Konstante.

Hinweis: Hashing

- Skizzieren Sie die Gestalt dieser Datenstruktur, wenn Sie der Reihe nach Datensätze mit den Schlüsseln 45, 46, ..., 50, 20, 62, 15, 24, 10, 9, 30, 56 und 67 einfügen.  
Verwenden Sie hierbei `h_size = 5` und die Hashfunktion  $h(s) = s \bmod \text{h\_size}$ .

**Aufgabe 28** ( $B$ -Bäume, 5T+2P)

a) Zeigen Sie:

- Ein  $B$ -Baum der Höhe  $h$  besitzt mindestens  $(\lceil B/2 \rceil)^{h-1}$  und höchstens  $B^h$  Blätter. (Achten Sie dabei auf die Sonderrolle des Wurzelknotens.)
- Ein  $B$ -Baum, in dem  $n$  Datensätze gespeichert sind, hat die Höhe  $h = \mathcal{O}(\log n)$ .

Verwenden Sie zum Beweis die erste Behauptung.

b) Skizzieren Sie nach jeder der folgenden Operationen den entstehenden  $B$ -Baum, wobei  $B = 4$  gelten soll:

- leeren Baum erzeugen
- Einfügen der Datensätze mit den Schlüsseln 60, 59, ..., 50 in dieser Reihenfolge
- Löschen des Datensatzes mit dem Schlüssel 50
- Einfügen der Datensätze mit den Schlüsseln 70 und 71
- Löschen des Datensatzes mit dem Schlüssel 58
- Löschen des Datensatzes mit dem Schlüssel 52

c) Ergänzen Sie die Datei `bb.c` um eine Funktion

```
bb_knoten *bb_max(b_baum b),
```

die einen Zeiger auf den Blattknoten mit maximalem Schlüssel zurückliefert. Ergänzen Sie außerdem das Hauptprogramm um den entsprechenden Menüpunkt.

**Abgabe:** Mi., 19.06.2002, 14 Uhr