

9. Übung:

Algorithmen und Datenstrukturen

Sommersemester 2008

13. Juni 2008

---

Abgabe bis Freitag, 20. Juni 2008, vor der Übung

**Aufgabe 9.1:** (Punkte 6)

AVL Baum

Entwickeln Sie den Hilfsalgorithmus  $Spalten(S_1, a, S_2, S_3)$  aus Vorlesung 16. Eine Beschreibung in Prosa reicht aus.

**Aufgabe 9.2:** (Punkte 4)

Union/Find

Implementieren Sie eine einfache Union/Find Datenstruktur als Klasse. Benutzen Sie einen (einzigen) Array (auf den Folien  $IST\_IN[]$  genannt). Die Schlüssel sind Zahlen aus dem Wertebereich  $[1..n]$ , die disjunkten Mengen werden über eindeutige Nummern identifiziert. D.h. die Schlüssel werden mit Hilfe des Arrays einer Menge zugeordnet, eine Unterscheidung zwischen internen und externen Mengen findet nicht statt. Drei Funktionen sind nötig:

- $init(n)$ : Initialisiert einen Array der Länge  $n$  (und  $n$  disjunkten Mengen).
- $find(x)$ : Liefert die Identität der Menge zurück in der  $x$  zu finden ist.
- $union(a, b)$ : Vereint die Mengen mit den Identitäten  $a$  und  $b$ .

**Aufgabe 9.3:** (Punkte 6)

Bereichssuche

Gegeben ist ein binärer Suchbaum. Implementieren Sie eine Funktion die alle Werte des Baumes, die innerhalb eines gegebenen Wertebereiches liegen, in einer Liste zurückgibt. Eine passende Liste steht Ihnen natürlich zur Verfügung. Die Knoten des Baumes sind wie folgt definiert:

```
bin_tree_node {
int value;
bin_tree_node* left_child;
bin_tree_node* right_child;
}
```

**Aufgabe 9.4:**

(Punkte 4)

$BB[\alpha]$  Baum

Fügen Sie die Zahlen 1 bis 10 in einen  $BB[\alpha]$  Baum ein für ein  $\alpha \leq 1/4$ . Erklären Sie alle Zwischenschritte.