



Aufgabenblatt 6

Abgabe: 03.06.06, 24.00 Uhr

Die erste Testklausur findet am Montag, 29.05.2006 in der Vorlesung statt. Die ersten beiden Aufgaben dieses Blattes dienen als zusätzliche Vorbereitung auf die Testklausur.

Alternativ können Sie die Zusammenfassungen der ersten zwei Aufgaben auch in meinem Büro (Raum 1.162), bis spätestens Donnerstag, 1. Juni 13 Uhr, abgeben.

Inhalt der Testklausur wird der bis jetzt behandelte Vorlesungsstoff des Sommersemesters sein.

- 1. Zusammenfassung I** (leicht–mittel) (schriftlich, 3 Punkte)
Fassen Sie alles zusammen, was Sie bisher über Suchbäume und im Speziellen über optimale Suchbäume gelernt haben. Die Zusammenfassung soll handschriftlich sein und auf maximal eine DIN-A4-Seite passen.
Das Erstellen der Zusammenfassung können Sie gerne gemeinsam in Ihrer Lerngruppe machen, es muss allerdings jeder Übungsteilnehmer eine eigene Zusammenfassung verfassen! Schreiben Sie Ihren Namen und Übungsgruppe auf die Seite, scannen Sie die Seite ein und geben Sie sie als JPEG-Datei im eClaus-System ab.
- 2. Zusammenfassung II** (mittel) (schriftlich, 3 Punkte)
Fassen Sie alles zusammen, was Sie bisher über AVL-Bäume gelernt haben. Die Zusammenfassung soll handschriftlich sein und auf maximal eine DIN-A4-Seite passen.
Das Erstellen der Zusammenfassung können Sie gerne gemeinsam in Ihrer Lerngruppe machen, es muss allerdings jeder Übungsteilnehmer eine eigene Zusammenfassung verfassen! Schreiben Sie Ihren Namen und Übungsgruppe auf die Seite, scannen Sie die Seite ein und geben Sie sie als JPEG-Datei im eClaus-System ab.
- 3. Mehrdeutigkeit optimaler Suchbäume** (mittel) (schriftlich, 1+3 Punkte)
Optimale Suchbäume sind im Allgemeinen nicht eindeutig, d.h., zu gegebenen Schlüsselwerten mit deren Zugriffshäufigkeiten gibt es mehrere verschiedene Suchbäume mit minimaler mittlerer Suchdauer.

 - a) Geben Sie ein Beispiel mit $n \geq 5$ Knoten und zugehörige Zugriffswahrscheinlichkeiten an, für die der optimale Suchbaum nicht eindeutig ist.
 - b) Schätzen Sie in O -Notation möglichst genau ab, wieviel verschiedene optimale Suchbäume es bei n Knoten maximal geben kann. (Hinweis: je nach Ansatz werden Sie dafür evtl. die Stirlingsche Formel $n! \approx \sqrt{2\pi n} \cdot n^n e^{-n}$ benutzen können)
- 4. Lernprogramm I** (mittel) (schriftlich, 6 Punkte)
Schreiben Sie ein Lern- und Übungsprogramm für AVL-Bäume in Ada mit folgender Funktionalität:

Ausgabe Ein zufällig erzeugter Baum wird ausgegeben (Blatt 2, Aufgabe 1).

Einfügen Ein zufälliger Knoten x soll eingefügt werden.

- a) Der Benutzer soll angeben, an welchem Knoten x angefügt wird.
- b) Der Benutzer soll angeben, an welchem Knoten die Balance verletzt ist.
- c) Der Benutzer soll angeben, welche Rotationsart ausgeführt werden muss.

Bemerkung: Falls Sie die Aufgabe 1 auf dem Aufgabenblatt 2 nicht oder nicht ganz richtig gelöst haben sollten, können Sie auf der Vorlesungsseite unter *Neuigkeiten* Lösungsvorschläge herunterladen und verwenden.

5. Lernprogramm II (mittel) (schriftlich, 4 Punkte)

Erweitern Sie das Programm aus Aufgabe 3 mit folgender Funktionalität:

Löschen Ein zufälliger Knoten x soll gelöscht werden.

- a) Dabei soll der Benutzer angeben, durch welchen Knoten der Knoten x ersetzt wird.
- b) Der Benutzer soll angeben, an welchem Knoten die Balance verletzt ist.
- c) Der Benutzer soll angeben, welche Rotationsart ausgeführt werden muss (bei Bedarf muss der Benutzer die letzten zwei Fragen mehrfach beantworten).

Tipp: Verwenden Sie `Get_Immediate(c)` für die Eingabe des Benutzers. Dabei wird ein Character c ohne Bildschirmecho von der Tastatur eingelesen. Auf diese Weise muss der Baum nicht nach jeder Eingabe neu gezeichnet werden.

6. B-Bäume (mittel) (schriftlich, 3 Punkte)

Angenommen Sie wollen 20 Millionen Datensätze in einem B-Baum abspeichern. Die Zugriffszeit Ihrer Festplatte beträgt 10 Millisekunden (es ist eine alte Festplatte) und die Größe eines Datensatzes ist 4KB. Außerdem haben Sie 100MB Hauptspeicher zur Verfügung.

Welche Ordnung m wählen Sie für den B-Baum, wenn Sie die Suchzeit im schlechtesten Fall minimieren wollen?