Technische Universität München
Institut für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr
Dr. Jens Ernst

WS 2003/2004
Übungsblatt 6
28. November 2003

Grundlegende Algorithmen

Letzter Abgabetermin: Freitag, 5. Dezember 2003 (nach der Vorlesung)

Aufgabe 1 (10 Punkte)

Wie in der Vorlesung gezeigt, hängt die Anzahl der Vergleiche durch den Quicksort-Algorithmus (bei einer Eingabe von n Elementen) vom Rang des in jedem Aufteilungsschritt gewählten Pivot-Elements ab. Im worst case ist dieser Rang immer extremal, d.h. minimal oder maximal, und im best case ist das Pivot-Element stets der Median. Stellen Sie für beide Fälle die Rekursionsgleichung für die Anzahl der Vergleiche auf und lösen Sie diese. *Hinweis:* Vergleichen Sie die best-case-Rekursionsgleichung mit der von Mergesort.

Aufgabe 2 (10 Punkte)

Modifizieren Sie die in Abschnitt 3.3 der Vorlesung[1] durch exponentielle Suche realisierte Funktion `is_element(key` k`)` so, daß sie nicht nur für besonders kleine Schlüssel k sondern *auch* für besonders große k effizienter ist als die binäre Suche: Es soll die Anzahl der Operationen für `is_element(`k`)` durch

$$O(\min\{\log(\text{Rang}(k)), \log(m - \text{Rang}(k))\})$$

beschänkt sein, wenn m die Anzahl der Elemente im Array ist. Für welche Suchelemente hat man nun den worst case?

Aufgabe 3 (10 Punkte)

Geben Sie drei verschiedene binäre Suchbäume minimaler Höhe an, die jeweils die Schlüsselwerte $1, 2, \ldots, 10$ enthalten. Begründen Sie, warum die Höhe Ihrer Bäume minimal ist.

Aufgabe 4 (10 Punkte)

Für binäre Suchbäume ist die `insert()`-Operation wie folgt realisiert: Der einzufügende Schlüssel k wird zunächst mit Hilfe von `is_element()` gesucht. Wird k nicht gefunden, so führt diese Suche zu einem Blatt. An dieses wird ein neues Blatt mit Schüssel k als linkes oder rechtes Kind angehängt.
Geben Sie eine Folge von `insert()`-Operationen – nacheinander angewendet auf einen anfangs nur den Schlüssel 1 enthaltenden binären Suchbaum – an, durch die ein *linear entarteter* (d.h. maximal hoher) binärer Suchbaum mit den Schlüsseln $1, 2, \ldots, 10$ entsteht.

$\boxed{\text{Termin und Ort der Midterm-Klausur: 12.12.2003, 15-18 Uhr, Raum MI HS1}}$

[1] Eine äquivalente Darstellung finden Sie in Kap. 4.2.3 des Buchs von V. Heun