Institut für Computergraphik und Algorithmen

Abteilung für Algorithmen und Datenstrukturen

TECHNISCHE UNIVERSITÄT WIEN

Prof. Petra Mutzel
Gunnar W. Klau
Gabriele Kodydek
Günther Raidl
René Weiskircher

Wintersemester 2001/2002

Prüfung zur Vorlesung
Algorithmen und Datenstrukturen 1
28. Januar 2002

a) Machen Sie bitte die folgenden Angaben in deutlicher Blockschrift:

Vorname: ______________________________ Nachname: ______________________________

Matrikelnummer: ______________________ Studienkennzahl: ______________________

b) Klausur soll gewertet werden für (nur ein Kreuz!):
   ○ VO-Prüfung Algorithmen und Datenstrukturen 1 (*default*)
   ○ VO-Prüfung Rekursive Prozeduren und flexible Datenstrukturen

c) Legen Sie während der Klausur Ihren Studentenausweis vor sich auf das Pult.

d) Schreiben Sie die Lösungen direkt auf das jeweilige Aufgabenblatt. Wenn Ihnen das Papier ausgeht, bitten Sie die Aufsicht um Nachschub. Es ist nicht erlaubt, eigenes Papier zu verwenden!

e) Denken Sie daran, dass keinerlei Hilfsmittel erlaubt sind – weder Taschenrechner, irgendwelche Unterlagen, Mobiltelefone, ...

VOR DER ABGABE AUSZUFÜLLEN:
Geben Sie bitte die Anzahl der zusätzlich abgegebenen Blätter an: __________

Resultat:

Aufgabe	A 1	A 2	A 3	A 4	A 5		Note
maximale Punktzahl	10	10	10	10	10	50	———
erreichte Punktzahl							

Viel Erfolg!

Aufgabe 1: O/Θ/Ω-Notation **(10 Punkte)**

a) 4 Punkte
Sei $f : \mathbb{N} \to \mathbb{R}^+$ eine Funktion.

Zeigen oder widerlegen Sie die folgende Aussage:

$$f(n) = O(\sqrt{2}) \Leftrightarrow f(n) = O(\pi)$$

b) 6 Punkte
Nehmen Sie an, die Worst–Case–Laufzeit eines Algorithmus berechnet sich durch

$$f(n) = \begin{cases} f(\frac{n}{2}) + n & \text{für } n \geq 2 \\ 0 & \text{für } n = 1 \end{cases}$$

Schätzen Sie $f(n)$ mit Hilfe der Θ–Notation ab. Nehmen Sie dabei an, dass n eine Zweierpotenz ist.

Aufgabe 2: Heapsort **(10 Punkte)**

Sortieren Sie das Feld

4 1 3 2 16 9 10 14 8 7

mit Heapsort. Beschreiben Sie, was Sie tun und visualisieren Sie den Heap für jeden einzelnen Schritt.

Aufgabe 3: What am I? **(10 Punkte)**

Gegeben sei ein binärer Suchbaum mit $n \geq 1$ Knoten und Wurzel r. Betrachten Sie folgenden Algorithmus[1]:

Algorithmus 1 whatami(r, n)

```
p = minimum(r);
Gib p.key aus;
für i = 1 bis n − 1 {
  p = successor(p);
  Gib p.key aus;
}
```

a) 2 Punkte
Was macht der Algorithmus?

b) 4 Punkte
Analysieren Sie die Laufzeit des Algorithmus mit Hilfe der Θ–Notation. Begründen Sie Ihre Antwort.

c) 4 Punkte
Geben Sie Pseudocode für einen *rekursiven* Algorithmus an, der die gleiche Ausgabe produziert.

[1] Die Funktionen minimum und successor sind wie in der Vorlesung definiert.

Aufgabe 4: B–Bäume **(10 Punkte)**

Beweisen Sie den folgenden Satz:
Die Anzahl der Blätter in einem B–Baum T ist um eins größer als die Anzahl der Schlüssel in T.

Aufgabe 5: Hashing **(10 Punkte)**

Gegeben sei die Zahlenfolge

$$19 \quad 7 \quad 30 \quad 12 \quad 5 \quad 15 \quad 29 \quad 10 \quad 3 \quad 23 \quad 4$$

und die beiden Hashfunktionen

$$\begin{aligned} h_1(k) &= k \mod 17 \\ h_2(k) &= (k \mod 5) + 1 \ . \end{aligned}$$

a) 7 Punkte
Fügen Sie die Elemente der Folge mit Hilfe von *Double Hashing* in eine Hashtabelle mit 11 Positionen, nummeriert von 0 bis 10, ein. Es muss für jede Zahl erkenntlich sein, wie sie zu ihrem Platz in der Hashtabelle kommt.

b) 3 Punkte
Warum wäre die Wahl

$$h_2(k) = k \mod 5$$

eine schlechte Wahl?