Universität Konstanz
Lehrstuhl für Praktische Informatik
Prof. Dr. D. Wagner / Sabine Cornelsen

Entwurf und Analyse von Algorithmen
WS 2000/2001

1. Übungsblatt

Ausgabe: 27. Oktober 2000 **Abgabe:** 3. November 2000, 10 Uhr
Die Bearbeitung in Zweiergruppen ist ausdrücklich erwünscht.

Aufgabe 1: **4 Punkte**

Entwickeln Sie einen rekursiven Algorithmus, der testet, ob eine gegebene Zahl in einer gegebenen sortierten Folge von Zahlen vorkommt. Notieren Sie den Algorithmus in Pseudocode und bestimmen Sie seine asymptotische Laufzeit im schlechtesten Fall.

Aufgabe 2: **4 Punkte**

Zeigen Sie mit Hilfe der Substitutionsmethode, daß eine Funktion, die für große n durch

$$T(n) = 2 \cdot T(\lfloor n/2 \rfloor + 17) + n$$

definiert ist, in $\mathcal{O}(n \log n)$ liegt. (Muß die Basis des Logarithmus näher spezifiziert werden?)

Aufgabe 3: **4 Punkte**

Finden Sie eine asymptotisch scharfe Schranke für die rekursiv definierte Funktion

$$T(n) = 2 \cdot T(\lfloor \sqrt{n} \rfloor) + 1.$$

Aufgabe 4: **2 Punkte**

Geben Sie den größten ganzzahligen Wert an, so daß ein Algorithmus mit der Laufzeit

$$T_B(n) = a \cdot T_B(n/4) + n^2$$

asymptotisch schneller ist, als ein Algorithmus mit der Laufzeit

$$T_A(n) = 7 \cdot T_A(n/2) + n^2.$$

Aufgabe 5: **2 Punkte**

Wenden Sie den Aufteilungs-Beschleunigunssatz an, um eine asymptotisch scharfe Schranke zu $T(n) = 4 \cdot T(n/2) + n^3$ zu bestimmen.