Prof. Dr. Ir. Joost-Pieter Katoen

Christian Dehnert, Friedrich Gretz, Benjamin Kaminski, Thomas Ströder

Allgemeine Hinweise:

- Die **Hausaufgaben** sollen in Gruppen von je **2-3 Studierenden** aus der **gleichen Kleingruppenübung (Tutorium)** bearbeitet werden. **Namen und Matrikelnummern** der Studierenden sind auf jedes Blatt der Abgabe zu schreiben. **Heften bzw. tackern Sie die Blätter!**
- Die **Nummer der Übungsgruppe** muss **links oben** auf das **erste Blatt** der Abgabe geschrieben werden. Notieren Sie die Gruppennummer gut sichtbar, damit wir besser sortieren können.
- Die Lösungen müssen **bis Mittwoch, den 13.05.2015 um 12:15 Uhr** in den entsprechenden Übungskasten eingeworfen werden. Sie finden die Kästen am Eingang Halifaxstr. des Informatikzentrums (Ahornstr. 55). Alternativ können Sie die Lösungen auch vor der Abgabefrist direkt bei Ihrer Tutorin/Ihrem Tutor abgeben.
- In Aufgaben, bei denen Sie Algorithmen implementieren sollen, dürfen Sie Ihre Lösung als Pseudo-Code abgeben. Abgaben in verbreiteten imperativen Sprachen wie Java oder C++ sind ebenfalls erlaubt.

Tutoraufgabe 1 (Rekursionsgleichungen):

Geben Sie die Rekursionsgleichung $T(d, l)$ für die Laufzeit des folgenden Algorithmus an. Gehen Sie dabei davon aus, dass die elementaren Operationen $\{+, -, \cdot, /, \sqrt{}\} \in \Theta(1)$ liegen.

```
float sierpinski(int depth, float length) {
  if (depth > 0) {
    return  sierpinski(depth-1, length/2) +
            sierpinski(depth-1, length/2) +
            sierpinski(depth-1, length/2);
  } else {
    // berechne die Flaeche des Dreiecks
    return √(length² − (length/2)²) ∗ (length/2);
  }
}
```

Hinweis: `http://de.wikipedia.org/wiki/Sierpinski-Dreieck`

Tutoraufgabe 2 (Substitutionsmethode):

Gegeben sei die folgende Rekursionsgleichung:

$$T(n) \quad = \quad \begin{cases} 4T(\frac{n}{2}) + \frac{n^2}{\log_2(n)}, & \text{falls } n > 1 \\ 1, & \text{sonst} \end{cases}$$

- Schätzen Sie mit Hilfe des Rekursionsbaumes eine (möglichst asymptotisch scharfe) obere Schranke für die Komplexitätsklasse der Laufzeit $T(n)$, d.h., geben Sie eine nicht-rekursive Funktion $f(n)$ mit $T(n) \in \mathcal{O}(f(n))$ an.
- Beweisen Sie mit Hilfe der Substitutionsmethode, dass $T(n) \in \mathcal{O}(f(n))$.

Hinweis: Die n–te Partialsumme der harmonischen Reihe ist definiert durch $H_n := \sum_{i=1}^{n} \frac{1}{i}$ und hat folgende Eigenschaften:

(i) $\forall n \geq 1.\ \log(n) < H_n$

(ii) $\forall n > 1.\ H_n - \log(n) \ < \ H_{n-1} - \log(n-1)$

Tutoraufgabe 3 (Variablentransformation):

Nutzen Sie die Methode der Variablentransformation um eine (asymptotisch scharfe) obere Schranke für die Rekursionsgleichung

$$T(n) = T(\sqrt[3]{n}) + 3$$

zu bestimmen.

Tutoraufgabe 4 (Master-Theorem):

Geben Sie für folgende Rekursionsgleichungen (mit Begründung) an, ob diese mit dem Master-Theorem gelöst werden können. Falls dies der Fall ist, geben Sie auch die resultierende Komplexitätsklasse an.

a) $T(n) = 4 \cdot T\left(\frac{n}{2}\right) + 5n$

b) $T(n) = 4 \cdot T\left(\frac{n}{4}\right) + 4n$

c) $T(n) = 2 \cdot T\left(\frac{n}{2}\right) + n^2$

d) $T(n) = 2 \cdot T(n)$

Aufgabe 5 (Rekursionsgleichungen): (1+3+3+3+3+5* Punkte)

a) Im Folgenden betrachten wir das Problem der Türme von Hanoi. Gegeben sind 3 Stäbe. Initial sind die Stäbe 2 und 3 leer, und auf dem ersten Stab befinden sich n Scheiben paarweise unterschiedlicher Größe. Die Scheiben sind dabei in aufsteigender Größe angeordnet, die größte Scheibe liegt also unten und die kleinste oben (siehe Abbildung 1).

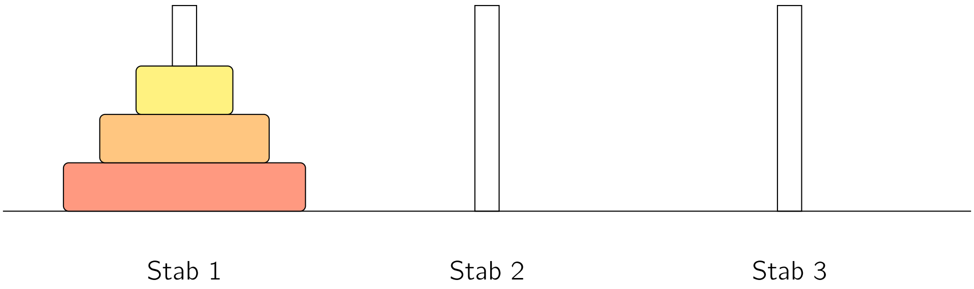

Abbildung 1: Die Türme von Hanoi mit $n = 3$ Scheiben.

Die Aufgabe ist nun, die Scheiben von Stab 1 auf Stab 3 zu verschieben. Dabei darf

- als elementare Operation immer nur die oberste Scheibe von einem Stab auf einen beliebigen anderen gelegt werden,
- niemals eine größere Scheibe auf einer kleineren liegen.

Ein Algorithmus, der eine das Problem löst, geht wie folgt vor:

```
// verschiebt die Scheiben von a nach c
// benutzt b als Puffer
verschiebe(int anzahlScheiben, Stab a, Stab b, Stab c) {
  if (anzahlScheiben > 0) {
    // verschiebe alle bis auf eine der Scheiben von a nach b
    verschiebe(anzahlScheiben - 1, a, c, b);

    // verschiebe oberste Scheibe von Stab a nach Stab c
    verschiebeObersteScheibe(a, c);

    // verschiebe die temporaer verschobenen Scheiben von b nach c
    verschiebe(anzahlScheiben - 1, b, a, c);
  }
}
```

Geben Sie die Rekursionsgleichung für die Laufzeit dieses Algorithmus in Abhängigkeit der Anzahl Scheiben n an.

b) Zeigen Sie mit Hilfe der Substitutionsmethode, dass für

$$T(n) = \begin{cases} 2 \cdot T(\lceil \frac{n}{2} \rceil) + n & \text{falls } n > 1 \\ 1 & \text{sonst} \end{cases}$$

gilt $T(n) \in \Omega(n \cdot \log n)$.

Hinweis: Es darf hierbei nicht davon ausgegangen werden, dass die Rundung beim rekursiven Aufruf einfach vernachlässigt werden darf.

c) Zeigen Sie, dass die Lösung der Rekursionsgleichung

$$T(n) = \begin{cases} T(n-1) + \frac{1}{n} & \text{falls } n > 1 \\ 1 & \text{falls } n = 1 \end{cases}$$

in $\Theta(\log n)$ liegt.

d) Zeigen Sie mit Hilfe der Rekursionsbaum-Methode, dass die Lösung der Rekursionsgleichung

$$T(n) = \begin{cases} T(\frac{n}{3}) + T(\frac{2}{3}n) + cn & \text{falls } n > 1 \\ 1 & \text{sonst} \end{cases}$$

in $\Omega(n \log n)$ liegt für alle $c \in \mathbb{R}$, $c > 0$.

e) Raten Sie mit Hilfe der Rekursionsbaum-Methode die Komplexitätsklasse (Θ) der Lösung der Rekursionsgleichung

$$T(n) = \begin{cases} T(\frac{n}{2}) + 2T(\frac{n}{6}) + n & \text{falls } n > 1 \\ 1 & \text{sonst} \end{cases}$$

und beweisen Sie ihre Vermutung anschließend mit Hilfe der Substitutionsmethode.

Hinweis: Um eine Vermutung herzuleiten, kann es hilfreich sein, die geometrische Summe anzuwenden:

$$\sum_{i=0}^{n} a_0 q^i = a_0 \cdot \frac{1 - q^{n+1}}{1 - q} \text{ für } q \neq 1.$$

f) Diese Aufgabe ist eine **Bonusaufgabe**. Das bedeutet, dass die Punkte nicht zur Summe der erreichbaren Punkte beitragen, aber in dieser Aufgabe erhaltene Punkte ganz normal gutgeschrieben werden.

Zeigen Sie mit Hilfe der Substitutionsmethode, dass für

$$T(n) = \begin{cases} 2 \cdot T(\lfloor \frac{n}{2} \rfloor) + n & \text{falls } n > 1 \\ 1 & \text{sonst} \end{cases}$$

gilt $T(n) \in \Omega(n \cdot \log n)$.
Hinweis: Es darf hierbei nicht davon ausgegangen werden, dass die Rundung beim rekursiven Aufruf einfach vernachlässigt werden darf.

Aufgabe 6 (Master-Theorem): (3+3+3+3 Punkte)

Geben Sie für folgende Rekursionsgleichungen (mit Begründung) an, ob diese mit dem Master-Theorem gelöst werden können. Falls dies der Fall ist, geben Sie auch die resultierende Komplexitätsklasse an.

a) $T(n) = 16 \cdot T\left(\frac{n}{4}\right) + 72n^{\sqrt{2}}$

b) $T(n) = 27 \cdot T\left(\frac{n}{9}\right) + n^2 + n^{\sqrt{2}}$

c) $T(n) = 16 \cdot T\left(\frac{n}{4}\right) + \frac{n^3+2n}{n}$

d) $T(n) = 4 \cdot T\left(\frac{n}{4}\right) + n \cdot \log_2(n)$