Gruppe $\boxed{\mathbf{14}}$
Datenstrukturen und Algorithmen

Phil Pützstück, 377247
Benedikt Gerlach, 376944
Sebastian Hackenberg, 377550

Hausaufgabe 5

Aufgabe 1

a)

Wir zeigen, dass für $n \in \mathbb{N}$ mit $n \geq 2$ stets $T(n) \in \mathcal{O}(n)$.

Sei $n \in [2, 15]_{\mathbb{N}}$. Dann ist

$$T(n) = 1 \leq n \log_2 n$$

Also existiert ein $c = 1 > 0$ sodass $T(n) \leq c \cdot n \log_2 n$.

Sei nun ein $n \in \mathbb{N}$ gegeben, sodass für alle $n' \in \mathbb{N}$ mit $n' < n$ ein $c > 0$ existiert für das $T(n) \leq c \cdot n \log_2 n$ ist (IV). Es folgt:

$$T(n) = 2 \cdot T\left(\frac{n}{4}\right) + T\left(\frac{n}{2}\right) + n \quad \overset{\text{IV}}{\leq} \quad 2c_1\left(\frac{n}{4}\log_2 \frac{n}{4}\right) + c_2\left(\frac{n}{2}\log_2 \frac{n}{2}\right) + n$$

$$= c_1\frac{n}{2}(\log_2 n - 2) + c_2\frac{n}{2}(\log_2 n - 1) + n \quad \leq \quad c\frac{n}{2}(\log_2 n - 2 + \log_2 n - 1) + n$$

$$= c\frac{n}{2}(2\log_2 n - 3) + n = cn\log_2 n - \frac{n}{2} \leq cn\log_2 n$$

wobei c_1, c_2 die Konstanten gemäß von (IV) sind und $c := \max\{c_1, c_2\}$. Folglich gilt nach dem Prinzip der vollständigen Induktion, dass es zu jedem $n \in \mathbb{N}, n \geq 2$ ein $c > 0$ gibt, sodass $T(n) \leq c \cdot n \log_2 n$ ist. Folglich ist $T(n) \in \mathcal{O}(n \log_2 n)$. $\square$

b)

Sei $n \in [1, 3]_{\mathbb{N}}$. Dann ist

$$T(n) = 1 \geq \frac{1}{3} \cdot 3^{\frac{n}{3}}$$

Also existiert ein $c = \frac{1}{3} > 0$ sodass $T(n) \geq c \cdot 3^{\frac{n}{3}}$.

Sei nun ein $n \in \mathbb{N}$ gegeben, sodass für alle $n' \in \mathbb{N}$ mit $n' < n$ ein $c > 0$ existiert für das $T(n) \geq c \cdot 3^{\frac{n}{3}}$ ist (IV). Es folgt:

$$T(n) = T(n-1) + T(n-2) + T(n-3) \quad \overset{\text{IV}}{\geq} \quad c_1 3^{\frac{n-1}{3}} + c_2 3^{\frac{n-2}{3}} + c_3 3^{\frac{n-3}{3}}$$

$$3c \cdot 3^{\frac{n-3}{3}} = c \cdot 3^{\frac{3}{n}}$$

wobei c_1, c_2, c_3 die Konstanten gemäß von (IV) sind und $c := \min\{c_1, c_2, c_3\}$. Folglich gilt nach dem Prinzip der vollständigen Induktion, dass es zu jedem $n \in \mathbb{N}$ ein $c > 0$ gibt, sodass $T(n) \geq c \cdot 3^{\frac{n}{3}}$ ist. Foglich ist $T(n) \in \Omega(3^{\frac{n}{3}})$. $\square$

Aufgabe 2

Aufgabe 3

a)

Der Rückgabewert des Aufrufs `T([3,1,2,4])` ist `[4,3,2,1]`.
Die 11 ersten `print`-Statements lauten wie folgt:

```
T: [3,1,2,4]
S: [3,1,2,4] [0,0,-1]
S: [1,2,4] [1,0,3]
S: [2,4] [2,0,3]
S: [4] [3,0,3]
S: [] [4,3,4]
T: [1,2,3]
S: [1,2,3] [0,0,-1]
S: [2,3] [1,0,1]
S: [3] [2,1,2]
S: [] [3,2,3]
```

b)

`S` gibt den Index des Maximums der Eingabeliste `L` zurück.
Dieser wird der Variable `i` dann in der Funktion `T` zugewiesen.

`c` stellt die bis jetzt größte gesehen Zahl dar; Die Funktion achtet nur auf den Kopf der Liste und rekursiert dann weiter, ist dieser Kopf größer als `c`, so ist `c` im rekursiven Aufruf eben dieser Kopf, also das momentane Maximum.

`a` stellt den Index von `L.head` in der ursprünglich übergebenen Liste `L` aus dem ersten Aufruf von `S` dar. Entsprechend wird `a` bei jedem rekursiven Aufruf inkrementiert, da wir beim rekursiven Aufrufe nur `L.tail` übergeben und damit über `L.head` in dem rekursiven Aufurf das nächste Element der Liste betrachten.

`b` ist dabei der Index von `c` in der urpsrünglich übergebenen Liste `L` aus dem ersten Aufruf von `S`, also der Index des momentanten Maximums `c`. Entsprechend wird `b` auf `a` gesetzt sobald auch `c` geändert wird, also ein neues Maximum gefunden wurde.

c)

Wenn wir die Aufrufe von S zählen ergeben sich die Rekursiongleichungen

$$T(n) = \begin{cases} 0 & n = 1 \\ S(n) + T(n-1) & \text{sonst} \end{cases} \qquad S(n) = \begin{cases} 0 & n = 0 \\ 1 + S(n-1) & \text{sonst} \end{cases}$$

Es ist offensichtlich $S(n) = n$:
Sei $n = 0$, dann ist $S(n) = 0 = n$. Sei $n \in \mathbb{N}_0$ mit $S(n) = n$ gegeben (IV).
Es folgt $S(n+1) = 1 + S(n) \overset{\text{IV}}{=} n + 1$. Folglich gilt nach Prinzip der vollständigen Induktion dass $S(n) = n$ für alle $n \in \mathbb{N}_0$. Es folgt also nun

$$T(n) = \begin{cases} 0 & n = 1 \\ n + T(n-1) & \text{sonst} \end{cases}$$

Damit gilt auch offensichtlich $T(n) = \sum_{i=1}^{n} i - 1 = \sum_{i=2}^{n} i$:
Sei $n = 1$, dann ist $T(n) = 0 \stackrel{\text{def}}{=} \sum_{i=2}^{1} i$. Sei $n \in \mathbb{N}$ mit $T(n) = \sum_{i=2}^{n} i$ gegeben (IV).
Es folgt

$$T(n+1) = n + 1 + T(n) \stackrel{\text{IV}}{=} n + 1 + \sum_{i=2}^{n} i = \sum_{i=2}^{n+1} i$$

Folglich gilt nach dem Prinzip der vollständigen Induktion dass $T(n) = \sum_{i=2}^{n} i$ für alle $n \in \mathbb{N}$.
Weiter gilt dann

$$T(n) = \sum_{i=2}^{n} i = \sum_{i=1}^{n} i - 1 = \frac{n(n+1)}{2} - 1 \in \Theta(n^2)$$

denn für $n_0 = 2$ gilt für alle $n \in \mathbb{N}$ mit $n \geq n_0$ dass

$$\frac{n(n+1)}{2} - 1 \leq n^2 \qquad \text{sowie} \qquad \frac{n(n+1)}{2} - 1 \stackrel{n \geq 2}{\geq} \frac{n^2}{2}$$

Also existieren auch $c_1 = 1$ und $c_2 = \frac{1}{2}$ sodass stets gilt

$$\forall n \in \mathbb{N}, n \geq n_0 : c_2 n^2 \leq T(n) \leq c_1 n^2$$

Folglich ist $T(n) \in \Theta(n^2)$. $\square$

Aufgabe 4