# Lösungsvorschläge zur 3. Übung

## Aufgabe 10

Es ist durch vollständige Induktion über $q$ für $n = a^q$ zu zeigen:

$$T(n) = \frac{c}{a} n \sum_{i=0}^{q} \left(\frac{b}{a}\right)^i$$

**Verankerung** $q = 0$

$$T(1) = \frac{c}{a} = \frac{c}{a} a^0 \sum_{i=0}^{0} \left(\frac{b}{a}\right)^i = T(q^0)$$

**Induktionsschritt** $q \Longrightarrow q+1$

$$\begin{aligned}
T(a^{q+1}) = T(aa^q) &\overset{\text{def}}{=} bT(a^q) + cn \\
&\overset{\text{I.-V.}}{=} b\frac{c}{a}a^q \left(\sum_{i=0}^{q} \left(\frac{b}{a}\right)^i\right) + ca^q = ca^q \left(\sum_{i=0}^{q} \frac{b}{a} \left(\frac{b}{a}\right)^i\right) + ca^q \\
&= ca^q \left(\left(\sum_{i=1}^{q+1} \left(\frac{b}{a}\right)^i\right) + 1\right) = \frac{c}{a} a^{q+1} \sum_{i=0}^{q+1} \left(\frac{b}{a}\right)^i
\end{aligned}$$

### 1. Fall $a > b$

Sei

$$K(n) := \sum_{i=0}^{\log_a n} \left(\frac{b}{a}\right)^i$$

Weil $\frac{b}{a} < 1$ ist, konvergiert die geometrische Reihe. Sei $\tilde{K} := \lim_{n\to\infty} K(n)$. Dann gilt, weil $K(n)$ monoton:

$$T(n) = \frac{c}{a} n K(n) \leq \frac{c}{a} n \tilde{K}$$

Also gilt $T(n) = O(n)$. Außerdem gilt für alle $n$: $K(n) \geq 1$, also auch

$$T(n) = \frac{c}{a} n K(n) \geq \frac{c}{a} n$$

Also ist auch $T(n) = \Omega(n)$ und demnach $T(n) = \Theta(n)$.

### 2. Fall $a = b$

$$\begin{aligned}
T(n) &= \frac{c}{a} n \sum_{i=0}^{\log_a n} 1^i = \frac{c}{a} n (1 + \log_a n) \\
&= \underbrace{\frac{c}{a} n}_{=:A} + \underbrace{\frac{c}{a} n \log_a n}_{=:B}
\end{aligned}$$

Für genügend große $n$ wächst $B$ schneller als $A$. Also braucht man nur $B$ betrachten. Demnach ist $T(n) = \Theta(n \log_a n)$. Aus Übung 1 weiß man, daß bei Laufzeiten die Basen der Logarithmen äquivalent sind, also ist $T(n) = \Theta(n \log_2 n)$.

### 3. Fall $a < b$

$$
\begin{aligned}
T(n) &= \frac{c}{a} a^q \sum_{i=0}^{q} \left(\frac{b}{a}\right)^i = \frac{c}{a} \sum_{i=0}^{q} b^i a^{q-i} = \frac{c}{a} \sum_{i=0}^{q} b^{q-i} a^i = \frac{c}{a} b^q \sum_{i=0}^{q} b^{-i} a^i \\
&= \frac{c}{a} b^{\log_a n} \underbrace{\sum_{i=0}^{\log_a n} \left(\frac{a}{b}\right)^i}_{=:K_2(n)}
\end{aligned}
$$

Aus den gleichen Überlegungen wie im ersten Fall konvergiert jetzt $K_2$ und $\tilde{K_2} := \lim_{n\to\infty} K_2(n)$. Also

$$
\begin{aligned}
T(n) &= \frac{c}{a} b^{\log_a n} K_2(n) = \frac{c}{a} n^{\log_a b} K_2(n) \\
&\leq \frac{c}{a} n^{\log_a b} \tilde{K_2}
\end{aligned}
$$

Also ist $T(n) = O(n^{\log_a b})$. Außerdem gilt für alle $n$, daß $K_2(n) \geq 1$.

$$
\begin{aligned}
T(n) &= \frac{c}{a} n^{\log_a b} K_2(n) \\
&\geq \frac{c}{a} n^{\log_a b}
\end{aligned}
$$

Demnach ist $T(n) = \Omega(n^{\log_a b})$ und auch $T(n) = \Theta(n^{\log_a b})$. Ufff...

# Aufgabe 11

## zu a)

Die Vorteile durch den Zeiger auf das Schwanz-Dummy und den Rückzeiger im Schwanz-Dummy sind:

- Einfügen an das *Ende* der Liste in $O(1)$.
- Verketten von zwei Listen in $O(1)$.

Beide Funktionen haben bei 'normalen' Listen die Laufzeit $O(n)$, da die gesamte Liste von Anfang bis Ende durchlaufen werden muß, um den Zeiger auf das letzte Element zu bekommen.

### zu b)

Der Vorteil ist: Normalerweise muß beim Einfügen und Löschen der 'Bauerntrick' angewandt werden, d.h. beim Einfügen werden die Daten des späteren Nachfolgers in das neu erzeugte Element umgespeichert. Beim Löschen werden die Daten des Nachfolgers in das zu löschende Element umgespeichert. Leider funktioniert der Bauerntrick nicht am *Ende* der Liste. Mit dem vorgeschlagenen Zeiger auf den Vorgänger funktionieren jetzt auch diese Spezialfälle. Allerdings bleiben bei 'normalen' Listen die Spezialfälle für das Löschen und Einfügen am *Anfang* der Liste, weil dann der Vorgänger des Elementes gar nicht existiert!