Übung 1

Shehata

1 Entfaltung + Beweisen

1.1 Lösen Sie folgende Rekurrenz mittels Entfaltung:

$$T(1) = 2$$
$$T(n) = T(\frac{n}{3}) + 3 \quad \text{for n} \geq 2$$

Resultat:

$$
\begin{aligned}
T(n) &= T(\frac{n}{3}) + 3 && i = 1 \\
T(n) &= (T((\frac{\frac{n}{3}}{3}) + 3) + 3 = T(\frac{n}{9}) + 6 && i = 2 \\
T(n) &= (T(\frac{\frac{n}{3}}{9}) + 6) + 3 = T(\frac{n}{27}) + 9 && i = 3 \\
T(n) &= (T(\frac{\frac{n}{3}}{27}) + 9) + 3 = T(\frac{n}{81}) + 12 && i = 4 \\
&T(\frac{n}{3^i}) + 3i
\end{aligned}
$$

$$
\begin{aligned}
3^i &= n && / \log \\
i \log(3) &= \log(n) && / : \log(3) \\
i &= \frac{\log(n)}{\log(3)} \\
i &= \log_3(n)
\end{aligned}
$$

$$
\begin{aligned}
&3\log_3(n) + T(\frac{n}{3^{\log_3(n)}}) && \\
&3\log_3(n) + T(\frac{n}{3^{\log_3(n)}}) && 3^{\log_3(n)} \Rightarrow n \\
&3\log_3(n) + T(\frac{n}{n}) && T(1) \Rightarrow 2 \\
&3\log_3(n) + 2 &&
\end{aligned}
$$

1.2 Um welche Art von Algorithmus handelt es sich hier? Interpretieren Sie die Rekurrenz

Die Rekursion drittelt mit jedem Aufruf die Datenmenge und f(n) ist immer 3.

1.3 Beweisen Sie (mittels Raten und Beweisen), dass Ihr Ergebnis aus 1.a) richtig ist

Assumption („wild guess"): $T(n) = 3\log_3(n) + 2$

Proof by Induction:
Base:

$$T(1) = 3\log_3(1) + 2 = 3 \times 0 + 2 = 2$$

Induction step:

$$
\begin{aligned}
T(n) &= T(\frac{n}{3}) + 3 \\
&= (3\log_3(\frac{n}{3}) + 2) + 3 \\
&= (3\log_3(\frac{n}{3}) + 2) + 3 \\
&= (3(\log_3(n) - \log_3(3)) + 2) + 3 \\
&= (3(\log_3(n) - 1) + 2) + 3 \\
&= (3\log_3(n) - 3 + 2) + 3 \\
&= 3\log_3(n) - 1 + 3 \\
&= 3\log_3(n) + 2
\end{aligned}
$$

2 Master Theorem

Lösen Sie folgende Rekurrenzen mit Hilfe des Master-Theorems:

(a) $T(n) = 4T(\frac{n}{2}) + 2n^2 + 4n$

(b) $T(n) = 27T(\frac{n}{3}) + n\log(n)$

(c) $T(n) = 8T(\frac{n}{2}) + \frac{n^8}{2} - 2n$

Geben Sie immer den Lösungsweg an und überprüfen Sie für den Fall, dass $f(n) = \Omega(n^{log_b(a+\varepsilon)})$ für $\varepsilon > 0$ (Fall 3), auch die Zusatzbedingung.

a)

$a = 4; \quad b = 2$
$f(n) = 2n^2 + 4n$
$\log_b(a) = \log_2(4) = 2$

1.

$$\begin{aligned} f(n) &= O(n^{\log_b(a)-\varepsilon}) \\ 2n^2 + 4n &= O(n^{2-\varepsilon}) && \rightarrow \text{Sum-Rule} \\ 2n^2 &\leq O(n^{2-\varepsilon}) \end{aligned}$$

Ist für $\varepsilon > 0$ immer kleiner, also falsch

2.

$$\begin{aligned} k &= 0 \\ f(n) &= \Theta(n^{2\log^k(n)}) \\ 2n^2 + 4n &= \Theta(n^2) && \rightarrow \text{Sum-Rule} \\ 2n^2 &= \Theta(n^2) \end{aligned}$$

Es wächst gleich schnell, also korrekt

Ergebnis: $\Theta(n\log_b(a)\log^{k+1}(n)) = \Theta(n^2\log^1(n)) = \Theta(n^2\log(n))$

b)

$a = 27; \quad b = 3$
$f(n) = n\log(n)\ \log_b(a) = \log_3(27) = 3$

1.

$$f(n) = O(n^{\log_3(27)-\varepsilon})$$
$$n\log(n) = O(n^{3-\varepsilon})$$
$$n\log(n) \leq O(n^{3-\varepsilon})$$
$$\rightarrow \text{Es wächst schneller für } \varepsilon = 1$$

Ergebnis: $\Theta(n^3)$

c)

$a = 8; \quad b = 2$
$f(n) = \frac{n^8}{2} - 2n$
$\log_b(a) = \log_2(8) = 3$

1.

$$f(n) = O(n^{\log_2(8)-\varepsilon})$$
$$\frac{n^8}{2} - 2n = O(n^{3-\varepsilon}) \qquad \rightarrow \text{Sum-Rule}$$
$$\frac{n^8}{2} \leq O(n^{3-\varepsilon})$$
$$\text{Ist für } \varepsilon > 0 \text{ immer kleiner, also falsch}$$

2.

$$k = 0$$
$$f(n) = \Theta(n^{3\log^k(n)})$$
$$\frac{n^8}{2} - 2n = \Theta(n^3) \qquad \rightarrow \text{Sum-Rule}$$
$$\frac{n^8}{2} = \Theta(n^3)$$
$$\text{Es wächst immer noch langsamer}$$

3.

$$f(n) = \Omega(n^{\log_2(8)+\varepsilon})$$
$$\frac{n^8}{2} - 2n = \Omega(n^{3+\varepsilon}) \qquad \rightarrow \text{Sum-Rule}$$
$$\frac{n^8}{2} \geq \Omega(n^{3+\varepsilon})$$

Die Aussage stimmt von $0 < \varepsilon < 5$

Zusatzbedingung

$$af(\frac{n}{b}) \leq cf(n) \tag{1}$$
$$8(\frac{(\frac{n}{2})^8}{2} - \frac{2n}{2}) \leq c\frac{n^8}{2} - 2n \tag{2}$$
$$8(\frac{\frac{n^8}{2^8}}{2} - n) \leq c\frac{n^8}{2} - 2n \tag{3}$$
$$8(\frac{n^8}{2^9} - n) \leq c\frac{n^8}{2} - 2n \tag{4}$$
$$\frac{8n^8}{8 * 2^9} - 8n \leq c\frac{n^8}{2} - 2n \tag{5}$$
$$\frac{n^8}{2^9} - 8n \leq c\frac{n^8}{2} - 2n \tag{6}$$
$$\frac{n^8}{2^9} \leq c\frac{n^8}{2} \tag{7}$$
$$\tag{8}$$

Ergebnis: $\Theta(n^8)$