

CS 307 Algorithmen und Datenstrukturen, Herbstsemester 2020

Übungsblatt 1

AUFGABE 1.1:

Füllen Sie die folgende Tabelle mit den Symbolen $O, o, \omega, \Omega, \Theta$ aus. Verwenden Sie dabei die Symbole O und Ω nur dann, wenn kein anderes Symbol verwendet werden kann.

	$\log n$	$2^{n/2}$	\sqrt{n}	5	2^n	n	e^n	n^2
$\log n$								
$2^{n/2}$								
\sqrt{n}								
5								
2^n								
n								
e^n								
n^2								

AUFGABE 1.2:

Finden Sie für jede der folgenden Funktionen $f_i : \mathbb{N}^{>1} \rightarrow \mathbb{R}^{\geq 0}$ eine möglichst kurz darstellbare Funktion $g_i : \mathbb{N}^{>1} \rightarrow \mathbb{R}^{\geq 0}$, so dass gilt: $f_i \in \Theta(g_i)$.

- $f_1(n) = n^2 2^n + 4^n + 3^n$
- $f_2(n) = n(n-1)/2$
- $f_3(n) = \log(n^{70})$
- $f_4(n) = 9n \log n + 30n(\log n)^2 + n$
- $f_5(n) = \sum_{i=1}^n 2^i$
- $f_6(n) = \frac{n}{\log n} + \frac{n^2}{\log^2 n}$

AUFGABE 1.3:

Seien f und g Funktionen von \mathbb{N} nach $\mathbb{R}^{\geq 0}$. Beweisen Sie oder finden Sie ein Gegenbeispiel:

- $f(n) + g(n) \in \Theta(\max\{f(n), g(n)\})$,

- Wenn $f(n) \notin O(g(n))$, so gilt: $f(n) \in \Omega(g(n))$.

AUFGABE 1.4: Versuchen Sie, für jede der folgenden Funktionen $T_i : \mathbb{N} \rightarrow \mathbb{R}$ eine möglichst genaue Beschreibung von $T_i(n)$ in der O-Notation anzugeben.

- a) $T_1(1) = 1$; $T_1(n) = 7T_1(\lceil n/2 \rceil) + n^2$ für $n > 1$,
- b) $T_2(1) = 1$; $T_2(n) = T_2(\lceil n/2 \rceil) + \sqrt{n}$ für $n > 1$,
- c) $T_3(1) = T_3(2) = 1$; $T_3(n) \leq 4T_3(\lfloor \sqrt{n} \rfloor) + \log^2(n)$ für $n > 2$.