

Landau-Symbole

Definition C.51

Eine Folge $A = \{a_n\}$ ist "Groß-O" von $B = \{b_n\}$, wenn der Quotient $\left\{\frac{a_n}{b_n}\right\}$ beschränkt ist. Die Folge A ist "Klein-o" von B , wenn $\left\{\frac{a_n}{b_n}\right\}$ eine Nullfolge ist. Wir schreiben $A = O(B)$ bzw. $A = o(B)$. O und o nennt man auch *Landau-Symbole*.

Beispiel C.52

- (a) $2n^2 + 3n + 4 = O(n^2)$
 (b) $2n^2 + 3n + 4 = o(n^3)$

Notation:

Im folgenden verknüpfen wir Folgen komponenten weise, so dass für $\circ \in \{+, -, *\}$ und $c \in \mathbb{R}$ eine Konstante:

$$A \circ B = \{a_n \circ b_n\}_{n \in \mathbb{N}} \quad \text{und} \quad c \cdot A = \{c \cdot a_n\}_{n \in \mathbb{N}}$$



Mehrdeutigkeit

Die Angabe $A = O(B)$ ist nicht eindeutig. Für $A = (n+2)$ sind z.B.:

$$\begin{aligned} A &= O(n^2), \\ A &= O(n), \\ A &= O\left(\frac{n}{2}\right) \end{aligned}$$

wahre Aussagen. Es gilt:

- ▶ Konstante Faktoren ändern nicht die Ordnung.
 $O(B) = O(c \cdot B), \quad \forall c \in \mathbb{R}$
- ▶ Terme niedriger Ordnung sind unwichtig:
 z.B. haben $\{n^2 + 3n + 4\}$ und $\{n^2 + 17n\}$ dieselbe Ordnung $O(n^2)$.



Satz C.53 (Rechenregeln für Landau-Symbole)

Für reelle Zahlenfolgen A, B, C und eine Konstante $c \in \mathbb{R}$ gilt:

- (a) $A = O(A), B = o(A) \Rightarrow B = O(A)$
 (b) $c \cdot A = O(A)$
 (c) $B = O(A) \wedge C = O(A) \Rightarrow B \pm C = O(A)$
 $B = o(A) \wedge C = o(A) \Rightarrow B \pm C = o(A)$
 (d) $A' = O(A) \wedge B' = O(B) \Rightarrow A' \cdot B' = O(A \cdot B)$
 (e) $\left. \begin{array}{l} A = O(B), B = O(C) \Rightarrow A = O(C) \\ A = o(B), B = o(C) \Rightarrow A = o(C) \end{array} \right\} \text{Transitivität}$



Nicht alle Folgen sind vergleichbar. Betrachte z.B. $A = \{a_n\}, B = \{b_n\}$ mit

$$a_n = \begin{cases} n^2 & (n \text{ gerade}) \\ n & (n \text{ ungerade}) \end{cases}$$

$$b_n = \begin{cases} n & (n \text{ gerade}) \\ n^2 & (n \text{ ungerade}) \end{cases}$$

Für gerades n sieht man, dass $A \neq O(B)$.
 Für ungerades n sieht man, dass $B \neq O(A)$.



Definition C.54

Wir sagen

$$O(A) = O(B) \Leftrightarrow A = O(B) \text{ und } B = O(A)$$

$$O(A) < O(B) \Leftrightarrow A = O(B) \text{ und } B \neq O(A)$$

Häufig verwendete Prototypen von Vergleichsfunktionen

O	Laufzeitverhalten
$O(1)$	konstant
$O(\log_a n), a > 1$	logarithmisch
$O(n)$	linear
$O(n \log_a n), a > 1$	$n \log n$
$O(n^2)$	quadratisch
$O(n^3)$	kubisch
$O(n^k)$	polynomial
$O(a^n)$	exponentiell



Es gilt:

$$O(1) < O(\log n) < O(n) < O(n \log n) < O(n^2) < O(n^3) < \dots < O(n^k) < \dots < O(2^n) < O(3^n) < \dots$$

n	$\text{ld } n$	$n \text{ ld } n$	n^2	n^3	2^n
10	3.32	33.22	100	1000	1024
100	6.64	664.4	10000	10^6	$1.27 \cdot 10^{30}$
1000	9.97	9966	10^6	10^9	$\approx 10^{301}$
10000	13.29	132877	10^8	10^{12}	$\approx 10^{3010}$

Für große n sollte man daher nicht auf Fortschritte im Rechnerbau hoffen, sondern nach Algorithmen suchen, die eine bessere Ordnung besitzen.

