



Figure 1: Überblick theoretische Informatik

1 Grundlagen

1.1 Notationen und begriffe

- \mathbb{N} bezeichnet die $\{1, 2, 3\}$
- \mathbb{N}_0 , sei $[n] = \{1, \dots, n\}$ und $[n]_0 = \{0, 1, \dots, n\}$
- Für eine Menge A und $n \in \mathbb{N}$ ist $A^n = \{(a_1, \dots, a_n) : a_1, \dots, a_n \in A\}$
- Für $n \in \mathbb{N}$ ist eine n -äre partielle funktion $\varphi : A^n \rightsquigarrow B$ eine Funktion mit $\text{dom}(\varphi) \subseteq A^n$ und $\text{Im}(\varphi) \subseteq B$. Für $a_1, \dots, a_n \in A$ bedeutet $\varphi(a_1, \dots, a_n) \downarrow$, dass $(a_1, \dots, a_n) \in \text{dom}(\varphi)$ gilt und $\varphi(a_1, \dots, a_n) \uparrow$ bedeutet, dass $(a_1, \dots, a_n) \notin \text{dom}(\varphi)$. Statt $\varphi(a_1, \dots, a_n) \uparrow$ schreiben wir auch $\varphi(a_1, \dots, a_n) = \uparrow$. Die partielle Funktion φ ist total, wenn $\text{dom}(\varphi) = A^n$ gilt.
- Eine lineare Ordnung, auch totale Ordnung, auf einer Menge A ist eine Relation $\leq \subseteq A^n$ sodass die folgende Eigenschaften erfüllt sind. (wie für Relationen üblich verwenden wir hier Infixnotation, schreiben also für $a, b \in A$ den Ausdruck $a \leq b$ anstatt $(a, b) \in \leq$):
 - (i) $a \leq a \ \forall a \in A$ (Reflexivität)
 - (ii) $a \leq b \wedge b \leq a \Rightarrow a = b \ \forall a, b \in A$ (Antisymmetrie)
 - (iii) $a \leq b, b \leq c \Rightarrow a \leq c \ \text{for all } a, b, c \in A$ (Transitivität)
 - (iv) $a \leq b \vee b \leq a \ \forall a, b \in A$ (Totalität)

1.2 Alphabet, Wörter und Sprachen

Eingaben und Ausgaben in unseren Berechnungsmodellen werden wörter genannt, wobei wir beliebige Zeichenketten als Wörter zulassen.

1.2.1 Definition (Alphabet)

Ein Alphabet ist eine nichtleere endliche Menge Σ . Das Alphabet Σ wird $|\Sigma|$ -är bezeichnet. Die Elemente von Σ heißen Buchstaben oder Symbole.

1.2.2 Definition (Wörter)

Ein Wort über einem Alphabet Σ ist eine endliche Folge von Symbolen aus Σ . Die Länge eines Wortes w ist $|w|$. Für $i \in |w|$ bezeichnet $w(i)$ das i -te Element von w und für Symbole $a_1, \dots, a_n \in \Sigma$ bezeichnet a_1, \dots, a_n das Wort w der Länge n mit $w(i) = a_i \forall i \in [n]$. Das Wort der Länge 0 heißt leeres Wort und wird λ bezeichnet. Ein Wort der Länge 1 wird mit dem Symbol $w(1)$ identifiziert.

1.2.3 Definition (Binäralphabet, Binärwörter)

Das Alphabet $\{0, 1\}$ heißt Binäralphabet. Die Wörter über dem Binäralphabet heißen Binärwörter.

1.2.4 Sprache

Eine **Sprache** ist eine Menge von Wörter über einem gemeinsamen Alphabet Σ . Einige einfache grundlegenden Sprachen sind die folgenden.

1.2.5 Definition

Die Menge aller Wörter über Σ wird mit Σ^* bezeichnet. Für $n \in \mathbb{N}_0$ setzen wir:

$$\Sigma^{\leq n} := \{w \in \Sigma^* : |w| \leq n\}$$

$$\Sigma^n := \{w \in \Sigma^* : |w| = n\}$$

$$\Sigma^{\geq n} := \{w \in \Sigma^* : |w| \geq n\}$$

$$\Sigma^+ := \Sigma^{\leq 1}$$

1.2.6 Definition (Verkettung)

Für Wörter w_1, w_2 ist die Verkettung $w_1 \circ w_2$, auch $w_1 w_2$, von w_1 und w_2 ist definiert durch:

$$w_1 \circ w_2 := w_1 \cdots w_1(|w_1|) w_2 \cdots w_2(|w_2|)$$

Für ein Wort w und $n \in \mathbb{N}_0$ ist w^k induktiv definiert durch $w^n := \lambda$ falls $n = 0$ und $w^n := w^{n-1} \circ w$ falls $n \geq 1$. Für eine Sprachen L_1, L_2 sei durch $L_1 \circ L_2$, auch $L_1 L_2$ definiert durch

$$L_1 \circ L_2 := \{w_1 w_2 : w_1 \in L_1, w_2 \in L_2\}$$

Für eine Sprache L und $n \in \mathbb{N}_0$ ist L^n moduliert definiert durch $L^n = \{\lambda\}$ falls $n = 0$ und $L^n := L \cdot L^{n-1}$ falls $n \geq 1$. Zudem sei $L^* := \bigcup_{n \in \mathbb{N}_0} L^n$. Für ein Wort w und eine Sprache L sei $wL := \{w\} \circ L$ und $Lw := L \circ \{w\}$.

Wir folgen der Konvention, dass \bullet^n und \bullet^* stärker binden als \circ ; für Wörter u, v gilt also $uv = u \circ (v^n)$. Insbesondere gilt auch $ab^n = a(b^n)$ für Symbole a, b eines Alphabets Σ .

1.2.7 Definition (Präfix, Infix, Suffix)

Seien u, v Wörter.

- (i) u ist Präfix von v , kurz $u \sqsubseteq v$, falls es ein Wort w gibt, sodass $uw = v$.
- (ii) u ist Infix von v falls es Wörter w_1, w_2 gibt sodass $v = w_1 u w_2$
- (iii) u ist Suffix von v , falls es ein Wort w gibt, sodass $v = wu$.

1.2.8 Definition (präfixfrei)

Eine Sprache heißt **präfixfrei**, wenn $u \sqsubseteq v \Rightarrow u = v \forall u, v \in L$.

1.2.9 Definition (Homomorphismus)

Für Sprache L und M heißt eine Funktion $\varphi : L \rightarrow M$ **Homomorphismus von Sprachen**, wenn $\varphi(uv) = \varphi(u)\varphi(v) \forall u, v \in L$ gilt.

1.2.10 Definition (Längenlexikographische Ordnung)

Ist Σ ein Alphabet und \leq eine lineare Ordnung auf Σ , so ist die zu \leq gehörige **längenlexikographische Ordnung** \leq_{lex} auf Σ^* die lineare Ordnung für die $u \leq_{lex} v$ genau dann für zwei verschiedene $u, v \in \Sigma^*$ gilt, wenn eine der folgenden Bedingungen gilt:

- $|u| < |v|$
- $|u| = |v|$ und ist $i \in [|u|]$ minimal mit $u(i) \neq v(i)$ so gilt $u(i) \leq v(i)$.

Bemerkung: Oft gehen wir von einer impliziten Ordnung auf Σ aus. Ist $\Sigma = a_1, \dots, a_n$ so gilt $a_1 \leq \dots \leq a_n$

1.2.11 Bemerkung

Sei Σ ein Alphabet $\forall w \in \Sigma^*$ ist $v \in \Sigma^* : v \leq_{lex} w$ endlich. Dies erlaubt es uns für ein Alphabet Σ die Wörter über Σ in längenlexilographischen Reihenfolge w_1, w_2, \dots zu betrachten, wobei wir w_i für $i \in \mathbb{N}$ als kleinstes Element von $\Sigma^* / w_1, \dots, w_{i-1}$ gewählt sei. Wir identifizieren oft \mathbb{N}_0 mit $0, 1^*$ indem wir $i \in \mathbb{N}_0$ mit in die längenlexilographische Reihenfolge $(i+1)$ -ten Wort $w_{i+1} \in 0, 1^*$ identifizieren.

$$\begin{array}{ccccc} \mathbb{N}_0 & 0 & 1 & 2 & 3 \\ \{0, 1\}^* & \lambda & 0 & 1 & 00 \end{array}$$

1.2.12 Definition

Es bezeichnet $bin : \mathbb{N}_0 \rightarrow \{0, 1\}^*$ die Funktion, für die $bin(i)$ das in längenlexikographischer Reihenfolge $(i+1)$ -te Binärwort ist $\forall i \in \mathbb{N}_0$

1.2.13 Bemerkung

$\forall i \in \mathbb{N}_0$ ist $1bin(i)$ die Binärdarstellung von $i+1$. Umgekehrt ist $\forall w \in 0, 1^*$ das $(2^{|w|} + \sum_{i \in [|w|]} w(i)2^{|w|-i})$ -te Binärwort.