

Lernzettel

Formale Sprachen, Alphabet, Wörter und Sprachen

Universität: Technische Universität Berlin
Kurs/Modul: Theoretische Grundlagen der Informatik
Erstellungsdatum: September 19, 2025



Zielorientierte Lerninhalte, kostenlos!
Entdecke zugeschnittene Materialien für deine Kurse:

<https://study.AllWeCanLearn.com>

Theoretische Grundlagen der Informatik

Lernzettel: Formale Sprachen, Alphabet, Wörter und Sprachen**(1) Alphabet**

Ein Alphabet Σ ist eine endliche, nichtleere Menge von Symbolen. Beispiel:

$$\Sigma = \{0, 1\}$$

Jedes Symbol $a \in \Sigma$ heißt Alphabetsymbol. Die Menge der endlichen Folgen von Symbolen aus Σ heißt Σ^* (Wörter bzw. Strings über Σ).

$$\Sigma^* = \bigcup_{i=0}^{\infty} \Sigma^i \quad \text{mit} \quad \Sigma^0 = \{\epsilon\}, \quad \Sigma^{i+1} = \Sigma \Sigma^i.$$

(2) Wörter (Strings)

Ein Wort über Σ ist ein Element von Σ^* . Es hat die Form

$$w = a_1 a_2 \dots a_m, \quad m \geq 0, \quad a_j \in \Sigma.$$

Beispiel: Für $\Sigma = \{0, 1\}$ gilt $w = 01011 \in \Sigma^*$. Der Ausdruck ϵ bezeichnet das leere Wort.

(3) Sprachen

Eine Sprache over Σ ist eine Teilmenge von Σ^* . Sie entspricht einer Menge von Wörtern.

$$L \subseteq \Sigma^*.$$

Beispiele: - Gerade Wörter über $\{0, 1\}$:

$$L_{\text{gerade}} = \{ w \in \{0, 1\}^* \mid |w| \text{ ist gerade} \}.$$

- Die Menge aller Wörter, die genau ein Symbol 0 enthalten:

$$L = \{ w \in \{0, 1\}^* \mid w \text{ enthält genau einmal } 0 \}.$$

(4) Operationen auf Sprachen

Seien $L_1, L_2 \subseteq \Sigma^*$.

- Konkatenation (Verkettung):

$$L_1 L_2 = \{ xy \mid x \in L_1, y \in L_2 \}.$$

- Vereinigung:

$$L_1 \cup L_2 = \{ w \mid w \in L_1 \text{ oder } w \in L_2 \}.$$

- Kleene-Stern:

$$L^* = \bigcup_{i=0}^{\infty} L^i, \quad L^0 = \{\epsilon\}, \quad L^{i+1} = L L^i.$$

- Spiegelung (Umkehr):

$$L^R = \{ w^R \mid w \in L \}, \quad w^R \text{ ist das Wort-Wende von } w.$$

(5) Formale Grammatiken (Grundidee)

Eine Grammatik G wird geschrieben als

$$G = (N, T, P, S),$$

mit - N : Menge von Nichtterminalsymbolen, - T : Menge von Terminalsymbolen (oft dieselbe Alphabetmenge wie Σ), - P : endliche Menge von Produktionsregeln $A \rightarrow \alpha$ mit $A \in N$ und $\alpha \in (N \cup T)^*$, - $S \in N$: Startsymbol.

Beispiel einer kontextfreien Grammatik, die die Sprache $\{a^n b^n \mid n \geq 0\}$ erzeugt:

$$G = (\{S\}, \{a, b\}, P, S), \quad P : S \rightarrow aSb \mid \epsilon.$$

Damit gilt $L(G) = \{a^n b^n \mid n \geq 0\}$.

(6) Chomsky-Hierarchie (kurz)

Die Hierarchie ordnet formale Sprachen nach der Art der Maschinen, die sie akzeptieren/erzeugen können:

- Typ-3 (reguläre Sprachen): akzeptiert von endlichen Automaten (DFA/NFA).

L ist regulär $\iff L$ wird durch einen endlichen Automaten erkannt.

- Typ-2 (kontextfreie Sprachen): erzeugt durch kontextfreie Grammatiken; erkannt von Kellerautomaten (Pushdown-Automat).

L ist kontextfrei $\iff L$ wird durch eine CFG erzeugt.

- Typ-1 (kontext-sensitive Sprachen): erzeugt von kontextsensitiven Grammatiken; erkannt von Linearbeschränkten Automaten.

- Typ-0 (unbeschränkte Sprachen): erzeugt durch allgemeine (unbeschränkte) Grammatiken; erkannt von Turingmaschinen.

Es gilt die Inklusion

$$\text{Typ-3} \subseteq \text{Typ-2} \subseteq \text{Typ-1} \subseteq \text{Typ-0}.$$

Hinweis zu Beispielen

- Die Sprache $\{a^n b^n \mid n \geq 0\}$ ist kontextfrei, aber nicht regulär.
- Die Sprache $\{a^i b^i c^i \mid i \geq 1\}$ ist weder regulär noch kontextfrei, sie liegt außerhalb der Typ-2-Sprachen.