

# Äquivalenz von Grammatiken

201

Seien  $G_1 = (V_1, \Sigma, P_1, S_1)$ ,  $G_2 = (V_2, \Sigma, P_2, S_2)$   
Grammatiken mit demselben Terminalalphabet  $\Sigma$ .

$G_1$  und  $G_2$  heißen **äquivalent**, falls  $\mathcal{L}(G_1) = \mathcal{L}(G_2)$ .

Beispiel: für  $\Sigma = \{a\}$

Grammatik  $G_1$ :

$$S_1 \rightarrow aS_1 \mid a$$

Grammatik  $G_2$ :

$$S_2 \rightarrow aA$$

$$A \rightarrow aA \mid \varepsilon$$

$$\mathcal{L}(G_1) = \mathcal{L}(G_2) = \{a^n : n \in \mathbb{N}, n \geq 1\}$$

**Typ 0:** beliebige Grammatik

**Typ 1:** kontextsensitive Grammatik:

alle Regeln haben die Form  $x \rightarrow y$  mit  $|x| \leq |y|$

**Typ 2:** kontextfreie Grammatik:

alle Regeln haben die Form  $A \rightarrow y$ , wobei  $A$  Variable

**Typ 3:** reguläre Grammatik:

alle Regeln haben die Form

$$A \rightarrow aB \quad A \rightarrow a$$

wobei  $A, B$  Variablen,  $a$  Terminal

$$A \rightarrow \varepsilon$$



induktive Definition vollständig geklammerter arithmetischer Ausdrücke gebildet aus

- Konstanten  $k \in \mathbb{N}$
- Variablen  $x_1, \dots, x_n$
- den Basisoperatoren

Addition  $+$

Multiplikation  $*$

Umformulierung  
als kontextfreie  
Grammatik

1. Jede Konstante  $k \in \mathbb{N}$  ist ein Ausdruck.
2. Jede Variable  $x_i$  ist ein Ausdruck,  $i = 1, \dots, n$ .
3. Mit  $\alpha, \beta$  sind auch  $(\alpha + \beta)$  und  $(\alpha * \beta)$  Ausdrücke.
4. Sonst nichts ist ein Ausdruck.

kontextfreie Grammatik  $G$  bestehend aus:

Alphabet  $\Sigma = \{+, *, (, ), x_1, \dots, x_n\} \cup \{0, 1, \dots, 9\}$

Nichtterminalen  $S, A, B$

$$S \rightarrow A \mid x_1 \mid \dots \mid x_n \mid (S + S) \mid (S * S)$$
$$A \rightarrow 0 \mid 1B \mid 2B \mid \dots \mid 9B$$
$$B \rightarrow \varepsilon \mid 0B \mid 1B \mid \dots \mid 9B$$

erzeugte Sprache:

$\mathcal{L}(G)$  = Menge aller Ausdrücke gemäß 1., 2., 3., 4.

für korrekt geklammerte arithmetische Ausdrücke

Alphabet:  $\Sigma = \{+, *, (, ), x_1, \dots, x_n\} \cup \{0, 1, \dots, 9\}$

Nichtterminale:  $S, T, U, A, B, C$

$S$	$\rightarrow$	$T$		$S + T$
$T$	$\rightarrow$	$U$		$T * U$
$U$	$\rightarrow$	$(S)$		$A$   $C$
$A$	$\rightarrow$	$0$		$1B$   $2B$   $\dots$   $9B$
$B$	$\rightarrow$	$\varepsilon$		$0B$   $1B$   $\dots$   $9B$
$C$	$\rightarrow$	$x_1$		$x_2$   $\dots$   $x_n$

$$\begin{aligned}
 S &\Rightarrow T \\
 &\Rightarrow T * U \\
 &\Rightarrow T * (S) \\
 &\Rightarrow T * (S + T) \\
 &\Rightarrow T * (T + T) \\
 &\vdots \\
 &\Rightarrow U * (U + U) \\
 &\vdots \\
 &\Rightarrow 7 * (x_2 + 25)
 \end{aligned}$$

Eine Grammatik  $G = (V, \Sigma, \mathcal{P}, S)$  heißt **kontextsensitiv**, falls einer der folgenden beiden Fälle zutrifft:

1. Fall:  $\varepsilon \notin \mathcal{L}(G)$  und alle Regeln haben die Form

$$x \rightarrow y \text{ mit } |x| \leq |y|$$

2. Fall:  $\varepsilon \in \mathcal{L}(G)$  und

(1)  $G$  enthält die Regel  $S \rightarrow \varepsilon$

(2) für alle anderen Regeln  $x \rightarrow y$  gilt:

$|x| \leq |y|$  und  $S$  kommt in  $y$  nicht vor

$S \rightarrow \varepsilon \mid Ab, A \rightarrow Ab \mid a$  kontextsensitiv

$S \rightarrow \varepsilon \mid Ab, A \rightarrow Sb$  nicht kontextsensitiv

Sei  $G = (V, \Sigma, P, S)$  eine CFG.  $G$  heißt  $\varepsilon$ -frei, falls gilt:

- (1) aus  $A \rightarrow \varepsilon$  folgt  $A = S$ .
- (2) falls  $S \rightarrow \varepsilon$ , so gibt es keine Regel  
 $A \rightarrow y$  mit  $y = \dots S \dots$

Fakt 1: Jede  $\varepsilon$ -freie CFG ist kontextsensitiv.

Fakt 2: Zu jeder CFG  $G$  gibt es eine äquivalente  $\varepsilon$ -freie CFG  $G'$ .

Folgerung:

Jede kontextfreie Sprache ist kontextsensitiv.

gegeben: CFG  $G = (V, \Sigma, \mathcal{P}, S)$

Ausgabe: äquivalente  $\varepsilon$ -freie CFG

1. berechne  $V_\varepsilon = \{A \in V : A \Rightarrow^* \varepsilon\}$

2. entferne alle  $\varepsilon$ -Regeln

3. solange es eine Regel  $B \rightarrow xAy$  gibt mit

$$A \in V_\varepsilon, |xy| \geq 1, B \not\rightarrow xy$$

wähle eine solche und füge die Regel  $B \rightarrow xy$  ein

4. falls  $S \in V_\varepsilon$ , dann füge ein neues Startsymbol  $S'$  und die Regeln  $S' \rightarrow S$  und  $S' \rightarrow \varepsilon$  ein.

gegeben: CFG  $G = (V, \Sigma, \mathcal{P}, S)$

Aufgabe: berechne  $\{A \in V : A \Rightarrow^* \varepsilon\}$

1. Markiere alle Nichtterminale  $A$  mit  $A \rightarrow \varepsilon$ .
2. Solange es eine Regel  $B \rightarrow A_1 \dots A_n$  gibt, wobei  $A_1, \dots, A_n$  markiert sind und  $B$  nicht markiert ist, wähle eine solche Regel und markiere  $B$ .
3. Gib alle markierten Nichtterminale aus.

Beispiel:

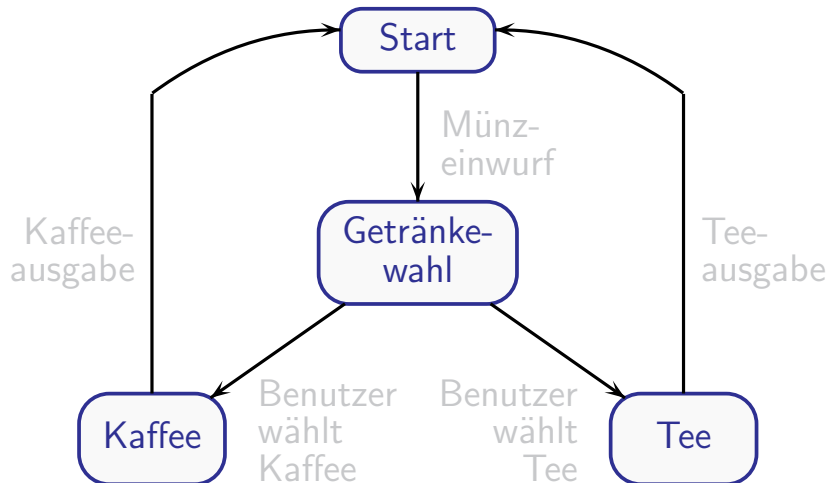
$S \rightarrow aB \mid \underline{BAA}$	$A \rightarrow bb \mid \underline{\varepsilon}$	$D \rightarrow aAS$
$B \rightarrow a \mid \underline{AC}$	$C \rightarrow DS \mid \underline{\varepsilon}$	

Sprachen vom Typ 0

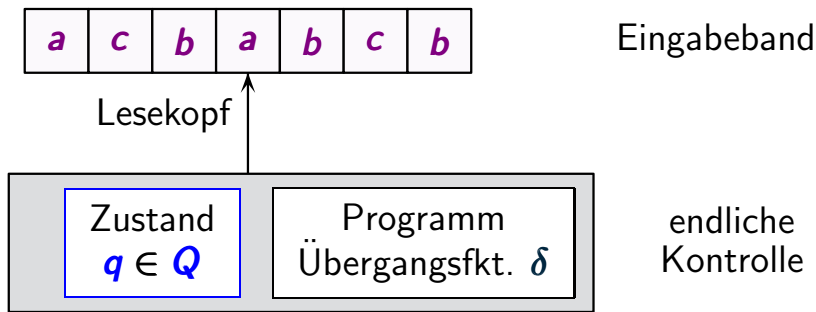
kontextsensitive Sprachen

kontextfreie Sprachen  
Kellerautomaten

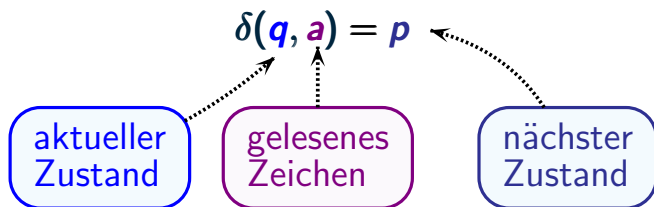
reguläre Sprachen  
endliche Automaten

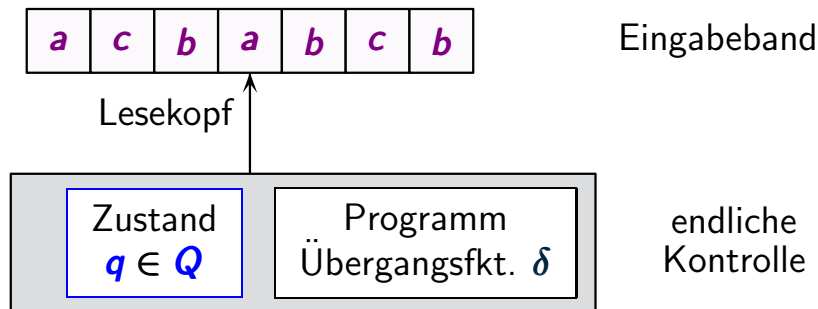


analog: Fahrkartenautomat, Geldautomat, ...



Übergangsfunktion: partielle Funktion  $\delta : Q \times \Sigma \rightarrow Q$





Übergangsfunktion: partielle Funktion  $\delta : Q \times \Sigma \rightarrow Q$

- die Eingabe wird von links nach rechts gelesen
- Zustandsübergänge gemäß  $\delta$
- nach Lesen der Eingabe: Akzeptanz oder Verwurf

Ein DFA ist ein Tupel  $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$

- endliche Menge  $Q$  von Zuständen
- Alphabet  $\Sigma$
- Anfangszustand  $q_0 \in Q$
- $F \subseteq Q$  Menge von Endzuständen
- partielle Übergangsfunktion  $\delta : Q \times \Sigma \rightarrow Q$

englische Abkürzung:

DFA für deterministic finite automaton

DFA  $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ , Wort  $w = a_1 \dots a_n \in \Sigma^*$

Der Lauf für  $w$  in  $\mathcal{M}$  ist die eindeutig bestimmte Zustandsfolge  $p_0 p_1 \dots p_m$ , für die gilt:

- (1)  $p_0 = q_0$  Anfangszustand
- (2)  $p_i = \delta(p_{i-1}, a_i)$  für  $1 \leq i \leq m$
- (3)  $m = |w| = n$  oder  $m < n$  &  $\delta(p_m, a_{m+1}) = \perp$

Der Lauf  $p_0 p_1 \dots p_m$  für  $w$  heißt

- akzeptierend, falls  $m = |w| = n$  und  $p_n \in F$
- verwerfend, falls  $p_0 p_1 \dots p_m$  nicht akzeptierend ist, also falls  $m < n$  oder  $m = n$  &  $p_n \notin F$