LMU
LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Formale Sprachen und Komplexität
Theoretische Informatik für Medieninformatiker
Sommersemester 2022

FSK & TIMI
Begrüßung, Organisatorisches, Inhaltsübersicht und Grundlagen

Prof. Dr. David Sabel

LFE Theoretische Informatik

Letzte Änderung der Folien: 26. April 2022

Personen

- **Dozent:**
  Prof. Dr. David Sabel
  Email: david.sabel@ifi.lmu.de
- **Wissenschaftliche Mitarbeiter:innen:**
  Sarah Vaupel
  Stephan Barth
- **Tutor:inn:en und Korrektor:inn:en:**
  Charlotte Gerhaher
  David Mosbach
  Elisabeth Lempa
  Elisabeth Schwertfellner
  Lea Korn
  Luca Maio
  Lukas Bartl
  Michael Fink Amores
  Simon Rossmair
  Thomas Grill

Zielgruppe der Veranstaltung (Hörerkreis)

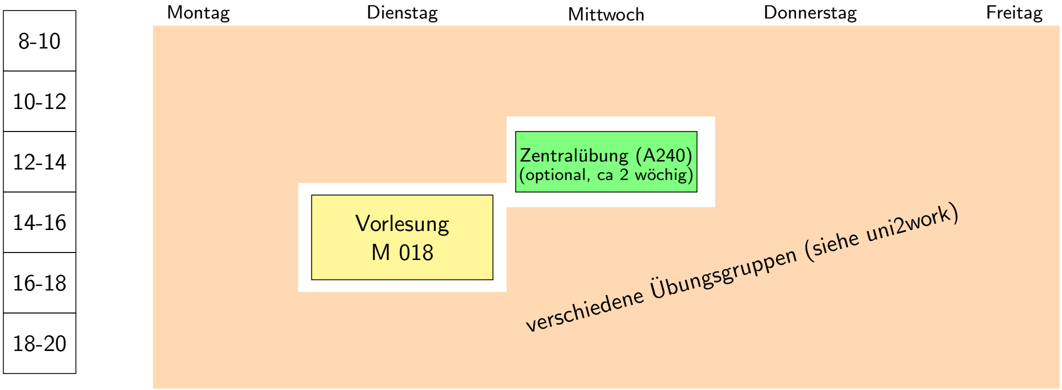
Formale Sprachen und Komplexität [FSK]:

- Studierende der Informatik
- Studierende der Bioinformatik
- Studierende im Lehramt
- Studierende im Nebenfach Informatik

Theoretische Informatik für Medieninformatiker [TIMI]:

- Studierende der Medieninformatik

Struktur der Veranstaltung



- **Vorlesung:** FSK: 3V, TIMI 2V (integriert, Plan auf Webseite)
- **Digitale Alternative**: ScreenCasts aus dem SoSe 2021
- **Zentralübung:** Zusatzangebot, Fragestunde & Beispiele (Plan auf Webseite) **Raum A 240**
- **Übungen:** präsenz oder online; Besprechung der Hausaufgaben; FSK: 2Ü, TIMI: 1Ü

Webseiten

Webseiten zu den Veranstaltungen:

www.tcs.ifi.lmu.de/lehre/ss-2022/fsk und **www.tcs.ifi.lmu.de/lehre/ss-2022/timi**

Anmeldung im Uni2Work:

- **uni2work.ifi.lmu.de/course/S22/IfI/FSK**
- **uni2work.ifi.lmu.de/course/S22/IfI/TIMI**

Anmeldung ist **notwendig** für:

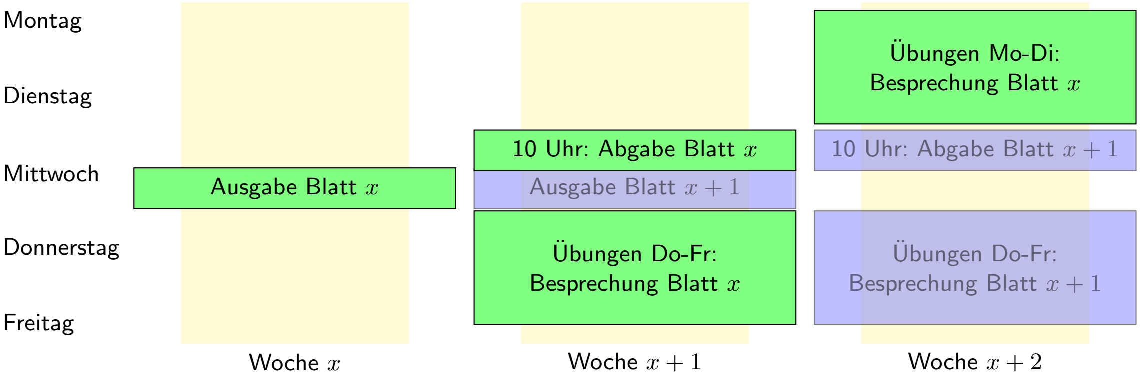
- Zugriff auf Material, Abgabe & Korrektur der Hausaufgaben
- Anmeldung zu den Übungsterminen
- Anmeldung zur Prüfung (noch nicht online)

Zulip-Chat

- Server-Adresse: **chat.ifi.lmu.de**
- Stream: **TCS-22S-FSK-TIMI**

Fragen und Kommentare am besten dort stellen.

Hausaufgaben



- Übungen diese Woche ab Donnerstag: Kennenlernen+Besprechung Blatt 0 (ohne Abgabe)
- Ausgabe, Abgabe und Korrektur elektronisch über Uni2Work
- Prüfungsbonus für erfolgreiches Bearbeiten der Aufgaben

Korrektur und Bonuspunkte

- Ausgewählte Hausaufgaben werden bepunktet
- Für jede Lösung zu einer bepunkteten Aufgabe gibt es 0 oder 1 oder 2 Punkte

Bonusregelung (gilt für Prüfung und Nachholprüfung im SoSe 2022):

100% der erreichbaren Übungspunkte entsprechen 10% der Prüfungspunkte

$$\textbf{Prüfungsbonus} = \frac{\text{erreichte Übungpunkte}}{\text{maximale Übungspunkte}} \cdot 0{,}1 \cdot \text{maximale Prüfungspunkte}$$

wenn die Prüfung bestanden ist (Bonuspunkte helfen nicht zum Bestehen)

Die Prüfung ist auf jeden Fall bestanden,
wenn 50% der Prüfungspunkte erreicht wurden.

Prüfungen

- **Plan (beantragt, noch nicht bestätigt):**
  Erstklausur am 17.08.2022 und Nachklausur am 21.09.2022
- Anmeldung zur Prüfung wird noch freigeschaltet
- Bonuspunkte gelten für Prüfung und Nachprüfung
- Teilnahme an der Nachholprüfung auch ohne Teilnahme an der Prüfung möglich

Material

- Vorlesungsfolien
- Skript zur Vorlesung (wird nach und nach bereit gestellt):
  Markierungen mit $\star$ für nicht-TIMI-relevante Teile
- ScreenCasts zur Vorlesung (aus SoSe 2021)
- Lehrbuch: Uwe Schöning, Theoretische Informatik – Kurz gefasst
- Hausaufgaben (Übungsblätter im Uni2Work)

Literatur

Wesentliche Quellen:

- Vorlesungsskript
- Uwe Schöning: Theoretische Informatik - kurz gefasst, 5. Auflage, Spektrum Akademischer Verlag, 2008 (ältere Auflagen sind auch in Ordnung)
  Teile sind u.U. zu kurz gefasst

Weitere Literatur:

- Alexander Asteroth und Christel Baier: Theoretische Informatik, Pearson Studium 2002.
  Gutes Buch, Aufbau in anderer Reihenfolge, Zugriff über UB
- John E. Hopcroft, Rajeev Motwani und Jeffrey D. Ullman: Introduction to Automata Theory, Languages, and Computation, 3. Auflage, 2006
  Der Klassiker, umfangreich (Erstauflage 1979!)
- Ingo Wegener: Theoretische Informatik - eine algorithmenorientierte Einführung, 3.Auflage, Teubner Verlag, 2005.
  Gutes Buch, andere Reihenfolge, Algorithmen stehen im Vordergrund, Zugriff über UB

Inhaltsübersicht über die Veranstaltung

Inhalte der Veranstaltung

Drei große wesentliche Themen der **Theoretischen Informatik**:

1. Formale Sprachen und Automatentheorie
   Wie stellt man Entscheidungsprobleme formal dar?

2. Berechenbarkeitstheorie
   Welche Probleme kann man algorithmisch (bzw. mit dem Computer) überhaupt lösen?

3. Komplexitätstheorie
   Welche Probleme kann man in annehmbarer Zeit lösen?

Inhalte: Formale Sprachen und Automatentheorie

- Chomsky-Grammatiken und Chomsky-Hierarchie
- Das Wortproblem und weitere Entscheidungsprobleme
- Reguläre Sprachen: reguläre Grammatiken, deterministische endliche Automaten, nichtdeterministische endliche Automaten, ε-Übergänge, reguläre Ausdrücke, Äquivalenz der Formalismen, Pumpinglemma, Satz von Myhill-Nerode, Minimalautomaten, Abschlusseigenschaften
- Kontextfreie Sprachen: kontextfreie Grammatiken, Chomsky-Normalform, Greibach-Normalform, Pumpinglemma, Ogden's Lemma, Cocke-Younger-Kasami-Algorithmus, Kellerautomaten, Abschlusseigenschaften
- Kontextsensitive Sprachen und Typ 0-Sprachen: kontextsensitive Grammatiken, Kuroda-Normalform, Turingmaschinen, Linear bounded automata (LBA), LBA-Probleme

TIMI: Zum Teil nur Auswahl der Inhalte / oberflächlichere Behandlung !

Inhalte: Berechenbarkeitstheorie

- Intuitive Berechenbarkeit, Churchsche These
- Turing-Berechenbarkeit, Varianten von Turingmaschinen (z.B. Mehrbandmaschinen)
- LOOP-, WHILE-, GOTO-Berechenbarkeit: LOOP-Programme WHILE-Programme, GOTO-Programme, Äquivalenz zu Turingmaschinen
- Primitiv-rekursive Funktionen, Ackermannfunktion, mu-Rekursion
- Halteproblem, Unentscheidbarkeit
- Rekursiv aufzählbar
- Reduktionen
- Postsches Korrespondenzproblem

TIMI: Zum Teil nur Auswahl der Inhalte / oberflächlichere Behandlung !

Inhalte: Komplexitätstheorie

- Zeitkomplexität
- Klassen P und NP
- NP-Härte, NP-Vollständigkeit
- polynomielle Reduktionen
- das SAT-Problem
- Satz von Cook
- weitere NP-vollständige Probleme (z.B. 3-SAT, Clique, Vertex Cover, Subset Sum, Knapsack, Directed Hamilition Circuit, Hamilition Circuit,. . .)

TIMI: Zum Teil nur Auswahl der Inhalte / oberflächlichere Behandlung !

Grundlagen: Worte und Formale Sprachen

Worte

Alphabet

Ein Alphabet Σ ist eine endliche nicht-leere Menge von Zeichen (oder Symbolen).

Z.B. $\Sigma = \{a, b, c, d, e\}$

Wort

Ein Wort w über Σ ist eine endliche Folge von Zeichen aus Σ.

Beispiele:

- $bade$ ist ein Wort über $\{a, b, c, d, e\}$
- $baden$ ist kein Wort über $\{a, b, c, d, e\}$

Weitere Notationen zu Worten

- Das leere Wort wird als ε notiert.
- Für $w = a_1 \cdots a_n$ ist $|w| = n$ die Länge des Wortes
- Für $1 \leq i \leq |w|$ ist $w[i]$ das Zeichen an i. Position in w.
- Für $a \in \Sigma$ und w ein Wort über Σ sei $\#_a(w) \in \mathbb{N}$ die Anzahl an Vorkommen des Zeichens a im Wort w

Beispiele:

- Es gilt $|\varepsilon| = 0$ und $\#_a(\varepsilon) = 0$ für alle $a \in \Sigma$.
- Für $\Sigma = \{a, b, c\}$ ist
  - $|abbccc| = 6$
  - $|aabbbccc| = 8$
  - $\#_a(abbccc) = 1$
  - $\#_c(aabbbccc) = 3$
- Für $w = abbbcd$ ist $w[1] = a$, $w[5] = c$ und $w[7]$ undefiniert.

Konkatenation und Kleene-Stern

Konkatenation

Das Wort uv (alternativ $u \circ v$) entsteht, indem Wort v hinten an Wort u angehängt wird.

Σ^* bezeichnet die Menge aller Wörter über Σ:

Definition von $\Sigma^i, \Sigma^*, \Sigma^+$

Sei Σ ein Alphabet, dann definieren wir:

$$\begin{array}{rcl}
\Sigma^0 & := & \{\varepsilon\} \\
\Sigma^i & := & \{aw \mid a \in \Sigma, w \in \Sigma^{i-1}\} \text{ für } i > 0 \\
\Sigma^* & := & \bigcup_{i \in \mathbb{N}} \Sigma^i \\
\Sigma^+ & := & \bigcup_{i \in \mathbb{N}_{>0}} \Sigma^i
\end{array}$$

Beachte: $\mathbb{N} = \{0, 1, 2, \ldots\}$ und $\mathbb{N}_{>0} = \{1, 2, \ldots\}$

Beispiele

Sei $\Sigma = \{a, b\}$.

Dann ist

$\Sigma^0 = \{\varepsilon\}$

$\Sigma^1 = \Sigma = \{a, b\}$

$\Sigma^2 = \{aa, ab, ba, bb\}$

$\Sigma^3 = \{\mathsf{x}w \mid \mathsf{x} \in \{a, b\}, w \in \Sigma^2\} = \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$

$\ldots$

und

$\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bba, bbb, aaaa, \ldots\}$

Beispiele

Sei $\Sigma = \{a, b\}$.

Dann ist

$\Sigma^0 = \{\varepsilon\}$
$\Sigma^1 = \Sigma = \{a, b\}$
$\Sigma^2 = \{aa, ab, ba, bb\}$
$\Sigma^3 = \{\mathsf{x}w \mid \mathsf{x} \in \{a, b\}, w \in \Sigma^2\} = \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$
$\ldots$

und

$\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bba, bbb, aaaa, \ldots\}$

Beispiele

Sei $\Sigma = \{a, b\}$.

Dann ist

$$\Sigma^0 = \{\varepsilon\}$$
$$\Sigma^1 = \Sigma = \{a, b\}$$
$$\Sigma^2 = \{aa, ab, ba, bb\}$$
$$\Sigma^3 = \{\mathsf{x}w \mid \mathsf{x} \in \{a, b\}, w \in \Sigma^2\} = \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$$
$$\ldots$$

und

$$\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bba, bbb, aaaa, \ldots\}$$

Beispiele

Sei $\Sigma = \{a, b\}$.

Dann ist

$$\Sigma^0 = \{\varepsilon\}$$
$$\Sigma^1 = \Sigma = \{a, b\}$$
$$\Sigma^2 = \{aa, ab, ba, bb\}$$
$$\Sigma^3 = \{\mathsf{x}w \mid \mathsf{x} \in \{a, b\}, w \in \Sigma^2\} = \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$$

$\cdots$

und

$$\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bba, bbb, aaaa, \ldots\}$$

Beispiele

Sei $\Sigma = \{a, b\}$.

Dann ist

$$\Sigma^0 = \{\varepsilon\}$$
$$\Sigma^1 = \Sigma = \{a, b\}$$
$$\Sigma^2 = \{aa, ab, ba, bb\}$$
$$\Sigma^3 = \{\mathsf{x}w \mid \mathsf{x} \in \{a, b\}, w \in \Sigma^2\} = \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$$

$\ldots$

und

$$\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bba, bbb, aaaa, \ldots\}$$

Beispiele

Sei $\Sigma = \{a, b\}$.

Dann ist

$$\Sigma^0 = \{\varepsilon\}$$
$$\Sigma^1 = \Sigma = \{a, b\}$$
$$\Sigma^2 = \{aa, ab, ba, bb\}$$
$$\Sigma^3 = \{\mathsf{x}w \mid \mathsf{x} \in \{a, b\}, w \in \Sigma^2\} = \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$$

$\ldots$

und

$$\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bba, bbb, aaaa, \ldots\}$$

Weitere Notationen und Begriffe

Sei w ein Wort über Σ

- w^m entsteht aus m-maligen Konkatenieren von w, d.h.

$$w^0 = \varepsilon \text{ und } w^m = ww^{m-1} \text{ für } m > 0$$

- $\overline{w}$ ist das rückwärts gelesene Wort w, d.h.

$$\overline{\varepsilon} = \varepsilon \text{ und für } w = a_1 \cdots a_n \text{ ist } \overline{w} = a_n a_{n-1} \cdots a_1$$

- w ist ein Palindrom g.d.w. $w = \overline{w}$

Beispiele für Palindrome: anna, reliefpfeiler, lagerregal, annasusanna

Sprechweisen: Präfix, Suffix, Teilwort

Seien u, v Wörter über einem Alphabet Σ.

- u ist ein **Präfix** von v, wenn es ein Wort w gibt mit $uw = v$.

$$\underbrace{\boxed{\quad u \quad}\boxed{\quad w \quad}}_{v}$$

- u ist ein **Suffix** von v, wenn es ein Wort w gibt mit $wu = v$.

$$\underbrace{\boxed{\quad w \quad}\boxed{\quad u \quad}}_{v}$$

- u ist ein **Teilwort** von v, wenn es Wörter w_1, w_2 gibt mit $w_1 u w_2 = v$.

$$\underbrace{\boxed{\quad w_1 \quad}\boxed{\quad u \quad}\boxed{\quad w_2 \quad}}_{v}$$

Beispiel: Sei $w = ababbaba$

- aba ist ein Präfix, Suffix und Teilwort von w
- $ababb$ ist ein Präfix (und Teilwort) von w, aber kein Suffix von w
- bab ist Teilwort von w, aber weder ein Präfix noch ein Suffix

Formale Sprache

Formale Sprache

Eine (formale) Sprache L über dem Alphabet Σ ist eine Teilmenge von Σ^* d.h. $L \subseteq \Sigma^*$

Beachte: Wir verwenden L für „language“.

Formale Sprache

Formale Sprache

Eine (formale) Sprache L über dem Alphabet Σ ist eine Teilmenge von Σ^* d.h. $L \subseteq \Sigma^*$

Beachte: Wir verwenden L für „language“.

Operationen auf formalen Sprachen

Seien L, L_1, L_2 formale Sprachen über Σ

- Vereinigung: $L_1 \cup L_2 := \{w \mid w \in L_1 \text{ oder } w \in L_2\}$
- Schnitt: $L_1 \cap L_2 := \{w \mid w \in L_1 \text{ und } w \in L_2\}$
- Komplement zu L: $\overline{L} := \Sigma^* \setminus L$
- Produkt: $L_1 L_2 = L_1 \circ L_2 = \{uv \mid u \in L_1 \text{ und } v \in L_2\}$

Beispiele

Sei $\Sigma = \{a, b\}$ und $L_1 = \{a^i \mid i \in \mathbb{N}\}$ $L_2 = \{b^i \mid i \in \mathbb{N}\}$.

- $L_1 \cup L_2 =$?
- $L_1 \cap L_2 =$?
- $\overline{L_1} =$?
- $L_1 L_2 =$?
- $L_2 L_1 =$?
- $L_1 L_1 =$?

Beispiele

Sei $\Sigma = \{a, b\}$ und $L_1 = \{a^i \mid i \in \mathbb{N}\}$ $L_2 = \{b^i \mid i \in \mathbb{N}\}$.

- $L_1 \cup L_2 =$ Sprache aller Wörter, die nur aus a's oder nur aus b's bestehen
- $L_1 \cap L_2 =$?
- $\overline{L_1} =$?
- $L_1 L_2 =$?
- $L_2 L_1 =$?
- $L_1 L_1 =$?

Beispiele

Sei $\Sigma = \{a, b\}$ und $L_1 = \{a^i \mid i \in \mathbb{N}\}$ $L_2 = \{b^i \mid i \in \mathbb{N}\}$.

- $L_1 \cup L_2 =$ Sprache aller Wörter, die nur aus a's oder nur aus b's bestehen
- $L_1 \cap L_2 = \{\varepsilon\}$
- $\overline{L_1} =$?
- $L_1 L_2 =$?
- $L_2 L_1 =$?
- $L_1 L_1 =$?

Beispiele

Sei $\Sigma = \{a, b\}$ und $L_1 = \{a^i \mid i \in \mathbb{N}\}$ $L_2 = \{b^i \mid i \in \mathbb{N}\}$.

- $L_1 \cup L_2 =$ Sprache aller Wörter, die nur aus a's oder nur aus b's bestehen
- $L_1 \cap L_2 = \{\varepsilon\}$
- $\overline{L_1} =$ Sprache der Worte, die mindestens ein b enthalten
- $L_1 L_2 =$?
- $L_2 L_1 =$?
- $L_1 L_1 =$?

Beispiele

Sei $\Sigma = \{a, b\}$ und $L_1 = \{a^i \mid i \in \mathbb{N}\}$ $L_2 = \{b^i \mid i \in \mathbb{N}\}$.

- $L_1 \cup L_2 =$ Sprache aller Wörter, die nur aus a's oder nur aus b's bestehen
- $L_1 \cap L_2 = \{\varepsilon\}$
- $\overline{L_1} =$ Sprache der Worte, die mindestens ein b enthalten
- $L_1 L_2 = \{a^i b^j \mid i, j \in \mathbb{N}\}$
- $L_2 L_1 =$?
- $L_1 L_1 =$?

Beispiele

Sei $\Sigma = \{a, b\}$ und $L_1 = \{a^i \mid i \in \mathbb{N}\}$ $L_2 = \{b^i \mid i \in \mathbb{N}\}$.

- $L_1 \cup L_2 =$ Sprache aller Wörter, die nur aus a's oder nur aus b's bestehen
- $L_1 \cap L_2 = \{\varepsilon\}$
- $\overline{L_1} =$ Sprache der Worte, die mindestens ein b enthalten
- $L_1L_2 = \{a^ib^j \mid i, j \in \mathbb{N}\}$
- $L_2L_1 = \{b^ia^j \mid i, j \in \mathbb{N}\}$
- $L_1L_1 =$?

Beispiele

Sei $\Sigma = \{a, b\}$ und $L_1 = \{a^i \mid i \in \mathbb{N}\}$ $L_2 = \{b^i \mid i \in \mathbb{N}\}$.

- $L_1 \cup L_2 =$ Sprache aller Wörter, die nur aus a's oder nur aus b's bestehen
- $L_1 \cap L_2 = \{\varepsilon\}$
- $\overline{L_1} =$ Sprache der Worte, die mindestens ein b enthalten
- $L_1 L_2 = \{a^i b^j \mid i, j \in \mathbb{N}\}$
- $L_2 L_1 = \{b^i a^j \mid i, j \in \mathbb{N}\}$
- $L_1 L_1 = L_1$

Für $L_1 = \{\spadesuit, \clubsuit, \diamondsuit, \heartsuit\}$ und $L_2 = \{7, 8, 9, 10, J, D, K, A\}$ stellt $L_1 L_2$ eine Repräsentation der Spielkarten eines Skatblatts dar.

Kleenescher Abschluss

Sei L eine Sprache. Dann ist:

$$L^0 := \{\varepsilon\}$$
$$L^i := L \circ L^{i-1} \text{ für } i > 0$$

$$L^* := \bigcup_{i \in \mathbb{N}} L^i$$
$$L^+ := \bigcup_{i \in \mathbb{N}_{>0}} L^i$$

Die Sprache L^* nennt man auch den Kleeneschen Abschluss von L
benannt nach Stephen Cole Kleene (1909-1994).

Kleenescher Abschluss

Sei L eine Sprache. Dann ist:

$$L^0 := \{\varepsilon\} \qquad L^* := \bigcup_{i \in \mathbb{N}} L^i$$

$$L^i := L \circ L^{i-1} \text{ für } i > 0 \qquad L^+ := \bigcup_{i \in \mathbb{N}_{>0}} L^i$$

Die Sprache L^* nennt man auch den Kleeneschen Abschluss von L benannt nach Stephen Cole Kleene (1909-1994).

Beispiel: $L = \{ab, ac\}$

- $L^0 = \{\varepsilon\}$
- $L^1 = L \circ L^0 = L = \{ab, ac\}$
- $L^2 = \{abab, abac, acab, acac\}$
- $L^3 = \{ababab, ababac, abacab, abacac, acabab, acabac, acacab, acacac\}$
- $L^* = \{\varepsilon\} \cup \{ax_1ax_2 \cdots ax_i \mid i \in \mathbb{N}_{>0}, x_j \in \{b, c\}, j = 1, \dots, i\}$.

Kleenescher Abschluss

Sei L eine Sprache. Dann ist:

$$L^0 := \{\varepsilon\} \qquad L^* := \bigcup_{i \in \mathbb{N}} L^i$$

$$L^i := L \circ L^{i-1} \text{ für } i > 0 \qquad L^+ := \bigcup_{i \in \mathbb{N}_{>0}} L^i$$

Die Sprache L^* nennt man auch den Kleeneschen Abschluss von L benannt nach Stephen Cole Kleene (1909-1994).

Beispiel: $L = \{ab, ac\}$

- $L^0 = \{\varepsilon\}$
- $L^1 = L \circ L^0 = L = \{ab, ac\}$
- $L^2 = \{abab, abac, acab, acac\}$
- $L^3 = \{ababab, ababac, abacab, abacac, acabab, acabac, acacab, acacac\}$
- $L^* = \{\varepsilon\} \cup \{ax_1ax_2 \cdots ax_i \mid i \in \mathbb{N}_{>0}, x_j \in \{b, c\}, j = 1, \ldots, i\}$.

Kleenescher Abschluss

Sei L eine Sprache. Dann ist:

$$L^0 := \{\varepsilon\} \qquad L^* := \bigcup_{i \in \mathbb{N}} L^i$$

$$L^i := L \circ L^{i-1} \text{ für } i > 0 \qquad L^+ := \bigcup_{i \in \mathbb{N}_{>0}} L^i$$

Die Sprache L^* nennt man auch den Kleeneschen Abschluss von L benannt nach Stephen Cole Kleene (1909-1994).

Beispiel: $L = \{ab, ac\}$

- $L^0 = \{\varepsilon\}$
- $L^1 = L \circ L^0 = L = \{ab, ac\}$
- $L^2 = \{abab, abac, acab, acac\}$
- $L^3 = \{ababab, ababac, abacab, abacac, acabab, acabac, acacab, acacac\}$
- $L^* = \{\varepsilon\} \cup \{ax_1ax_2 \cdots ax_i \mid i \in \mathbb{N}_{>0}, x_j \in \{b, c\}, j = 1, \ldots, i\}$.

Kleenescher Abschluss

Sei L eine Sprache. Dann ist:

$$L^0 := \{\varepsilon\} \qquad L^* := \bigcup_{i \in \mathbb{N}} L^i$$

$$L^i := L \circ L^{i-1} \text{ für } i > 0 \qquad L^+ := \bigcup_{i \in \mathbb{N}_{>0}} L^i$$

Die Sprache L^* nennt man auch den Kleeneschen Abschluss von L benannt nach Stephen Cole Kleene (1909-1994).

Beispiel: $L = \{ab, ac\}$

- $L^0 = \{\varepsilon\}$
- $L^1 = L \circ L^0 = L = \{ab, ac\}$
- $L^2 = \{abab, abac, acab, acac\}$
- $L^3 = \{ababab, ababac, abacab, abacac, acabab, acabac, acacab, acacac\}$
- $L^* = \{\varepsilon\} \cup \{ax_1 ax_2 \cdots ax_i \mid i \in \mathbb{N}_{>0}, x_j \in \{b, c\}, j = 1, \ldots, i\}$.

Kleenescher Abschluss

Sei L eine Sprache. Dann ist:

$$L^0 := \{\varepsilon\} \qquad L^* := \bigcup_{i\in\mathbb{N}} L^i$$

$$L^i := L \circ L^{i-1} \text{ für } i > 0 \qquad L^+ := \bigcup_{i\in\mathbb{N}_{>0}} L^i$$

Die Sprache L^* nennt man auch den Kleeneschen Abschluss von L benannt nach Stephen Cole Kleene (1909-1994).

Beispiel: $L = \{ab, ac\}$

- $L^0 = \{\varepsilon\}$
- $L^1 = L \circ L^0 = L = \{ab, ac\}$
- $L^2 = \{abab, abac, acab, acac\}$
- $L^3 = \{ababab, ababac, abacab, abacac, acabab, acabac, acacab, acacac\}$
- $L^* = \{\varepsilon\} \cup \{ax_1ax_2\cdots ax_i \mid i \in \mathbb{N}_{>0}, x_j \in \{b, c\}, j = 1, \ldots, i\}$.

Kleenescher Abschluss

Sei L eine Sprache. Dann ist:

$$L^0 := \{\varepsilon\} \qquad L^* := \bigcup_{i \in \mathbb{N}} L^i$$

$$L^i := L \circ L^{i-1} \text{ für } i > 0 \qquad L^+ := \bigcup_{i \in \mathbb{N}_{>0}} L^i$$

Die Sprache L^* nennt man auch den Kleeneschen Abschluss von L benannt nach Stephen Cole Kleene (1909-1994).

Beispiel: $L = \{ab, ac\}$

- $L^0 = \{\varepsilon\}$
- $L^1 = L \circ L^0 = L = \{ab, ac\}$
- $L^2 = \{abab, abac, acab, acac\}$
- $L^3 = \{ababab, ababac, abacab, abacac, acabab, acabac, acacab, acacac\}$
- $L^* = \{\varepsilon\} \cup \{ax_1ax_2 \cdots ax_i \mid i \in \mathbb{N}_{>0}, x_j \in \{b, c\}, j = 1, \ldots, i\}$.

Weitere Beispiele

$$((\{\varepsilon, 1\} \circ \{0, \dots, 9\}) \cup (\{2\} \circ \{0, 1, 2, 3\})) \circ \{:\} \circ \{0, 1, 2, 3, 4, 5\} \circ \{0, \dots, 9\}$$

Beschriebene Sprache = ?

$$\{0\} \cup (\{1, \dots, 9\} \circ \{0, \dots, 9\}^*)$$

Beschriebene Sprache = ?

Weitere Beispiele

$$((\{\varepsilon, 1\} \circ \{0, \ldots, 9\}) \cup (\{2\} \circ \{0, 1, 2, 3\})) \circ \{:\} \circ \{0, 1, 2, 3, 4, 5\} \circ \{0, \ldots, 9\}$$

Beschriebene Sprache = Sprache aller gültigen Uhrzeiten

$$\{0\} \cup (\{1, \ldots, 9\} \circ \{0, \ldots, 9\}^*)$$

Beschriebene Sprache = ?

Weitere Beispiele

$$((\{\varepsilon, 1\} \circ \{0, \ldots, 9\}) \cup (\{2\} \circ \{0, 1, 2, 3\})) \circ \{:\} \circ \{0, 1, 2, 3, 4, 5\} \circ \{0, \ldots, 9\}$$

Beschriebene Sprache = Sprache aller gültigen Uhrzeiten

$$\{0\} \cup (\{1, \ldots, 9\} \circ \{0, \ldots, 9\}^*)$$

Beschriebene Sprache = Sprache aller natürlichen Zahlen

Grammatiken und die Chomsky-Hierarchie

Prof. Dr. David Sabel

LFE Theoretische Informatik

Letzte Änderung der Folien: 26. April 2022

Formale Sprachen darstellen

- Sei Σ ein Alphabet.
- Eine Sprache über Σ ist eine Teilmenge von Σ^*.
- Z.B. für $\Sigma = \{(,),+,-,*,/,a\}$ sei
  L_{ArEx} die Sprache aller korrekt geklammerten Ausdrücke

  Z.B. $((a+a)-a)*a \in L_{ArEx}$ aber $(a-)+a) \notin L_{ArEx}$
- Unsere bisherigen Operationen auf Sprachen (Mengen) können das nicht darstellen

Benötigt: Formalismus, um L_{ArEx} zu beschreiben

Formale Sprachen darstellen (2)

Anforderungen:

- **Endliche** Beschreibung
- Sprache selbst muss aber auch unendlich viele Objekte erlauben

Zwei wesentliche solchen Formalismen sind

- Grammatiken
- Automaten

Grammatiken

Grammatik für einen sehr kleinen Teil der deutschen Sprache:

$$
\begin{aligned}
<\text{Satz}> &\rightarrow <\text{Subjekt}><\text{Prädikat}><\text{Objekt}> \\
<\text{Subjekt}> &\rightarrow <\text{Artikel}><\text{Attribut}><\text{Nomen}> \\
<\text{Objekt}> &\rightarrow <\text{Artikel}><\text{Attribut}><\text{Nomen}> \\
<\text{Artikel}> &\rightarrow \varepsilon \\
<\text{Artikel}> &\rightarrow \text{der} \\
<\text{Artikel}> &\rightarrow \text{das} \\
<\text{Attribut}> &\rightarrow <\text{Adjektiv}> \\
<\text{Attribut}> &\rightarrow <\text{Adjektiv}><\text{Attribut}> \\
<\text{Adjektiv}> &\rightarrow \text{kleine} \\
<\text{Adjektiv}> &\rightarrow \text{große} \\
<\text{Adjektiv}> &\rightarrow \text{nette} \\
<\text{Adjektiv}> &\rightarrow \text{blaue} \\
<\text{Nomen}> &\rightarrow \text{Mann} \\
<\text{Nomen}> &\rightarrow \text{Auto} \\
<\text{Prädikat}> &\rightarrow \text{fährt} \\
<\text{Prädikat}> &\rightarrow \text{liebt}
\end{aligned}
$$

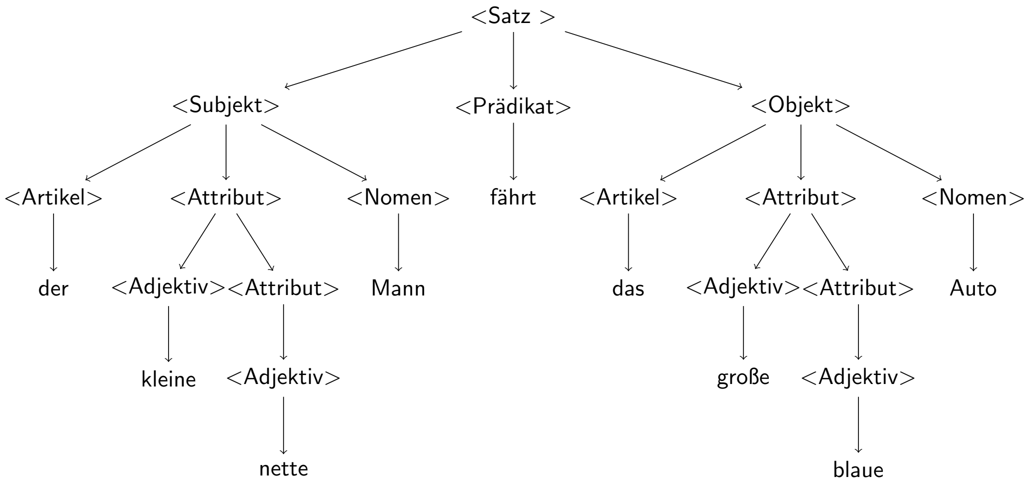
Grammatiken

- Endliche Menge von Regeln „linke Seite $\rightarrow$ rechte Seite"
- Symbole in spitzen Klammern wie $<$Artikel$>$ sind Variablen,
  d.h. sie sind Platzhalter, die weiter ersetzt werden müssen.
- Z.B. kann

  „der kleine nette Mann fährt das große blaue Auto"

  durch die obige Grammatik abgeleitet werden

Syntaxbaum zum Beispiel



Definition einer Grammatik

Definition (Grammatik)

Eine **Grammatik** ist ein 4-Tupel $G = (V, \Sigma, P, S)$ mit

- V ist eine endliche Menge von **Variablen**
  (alternativ Nichtterminale, Nichtterminalsymbole)
- Σ (mit $V \cap \Sigma = \emptyset$) ist ein **Alphabet** von **Zeichen**
  (alternativ **Terminale**, Terminalsymbole)
- P ist eine endliche Menge von **Produktionen** von der Form $\ell \to r$ wobei $\ell \in (V \cup \Sigma)^+$ und $r \in (V \cup \Sigma)^*$
  (alternativ Regeln)
- $S \in V$ ist das **Startsymbol**
  (alternativ Startvariable)

Manchmal genügt es, P alleine zu notieren
(wenn klar ist, was Variablen, Zeichen und Startsymbol sind)

Beispiel für eine Grammatik

$G = (V, \Sigma, P, E)$ mit

$$V = \{E\ , M, Z\},$$

$$\Sigma = \{+\ , *, 1, 2, (,)\}$$

$$\begin{aligned} P = \{E &\rightarrow M, \\ E &\rightarrow E + M, \\ M &\rightarrow Z, \\ M &\rightarrow M * Z, \\ Z &\rightarrow 1, \\ Z &\rightarrow 2, \\ Z &\rightarrow (E)\} \end{aligned}$$

Ableitung

Sei $G = (V, \Sigma, P, S)$ eine Grammatik.

Ableitungsschritt $\Rightarrow_G$

Für Satzformen u, v (d.h. Worte aus $(V \cup \Sigma)^*$) sagen wir:

u geht unter Grammatik G unmittelbar in v über, $u \Rightarrow_G v$, wenn

$$u = w_1 \ell w_2 \Rightarrow_G w_1 r w_2 = v \text{ mit } (\ell \to r) \in P$$

- Wenn G klar ist, schreiben wir $u \Rightarrow v$ statt $u \Rightarrow_G v$
- $\Rightarrow_G^*$ sei die reflexiv-transitive Hülle von $\Rightarrow_G$

Ableitung

Eine Folge $(w_0, w_1, \ldots, w_n)$ mit $w_0 = S$, $w_n \in \Sigma^*$ und $w_{i-1} \Rightarrow w_i$ für $i = 1, \ldots, n$ heißt Ableitung von w_n. Statt $(w_0, \ldots, w_n)$ schreiben wir auch $w_0 \Rightarrow \ldots \Rightarrow w_n$

Beispiel

$$G = (V, \Sigma, P, E) \text{ mit } V = \{E, M, Z\} \text{ und } \Sigma = \{+, *, 1, 2, (,)\} \quad \text{und}$$

$$\begin{array}{rlllllll} P = \{ E \rightarrow M, & E \rightarrow E + M, & M \rightarrow Z, & M \rightarrow M * Z, \\ Z \rightarrow 1, & Z \rightarrow 2, & Z \rightarrow (E) & \} \end{array}$$

Eine Ableitung von (2+1) * (2+2):

$$E \Rightarrow M$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$
\begin{aligned}
P = \{ E &\to M, & E &\to E + M, & M &\to Z, & M &\to M * Z, \\
Z &\to 1, & Z &\to 2, & Z &\to (E) & & \}
\end{aligned}
$$

Eine Ableitung von (2+1) * (2+2):

$$E \Rightarrow M \Rightarrow M * Z$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$\begin{array}{llllllllllll} P = \{E & \to & M, & E & \to & E + M, & M & \to & Z, & M & \to & M * Z, \\ Z & \to & 1, & Z & \to & 2, & Z & \to & (E) & & & \} \end{array}$$

Eine Ableitung von (2+1) * (2+2):

$$E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z$$

Beispiel

$$G = (V, \Sigma, P, E) \text{ mit } V = \{E, M, Z\} \text{ und } \Sigma = \{+, *, 1, 2, (,)\} \quad \text{ und}$$

$$\begin{array}{llllllllllll} P = \{E & \to & M, & E & \to & E + M, & M & \to & Z, & M & \to & M * Z, \\ Z & \to & 1, & Z & \to & 2, & \color{red}{Z} & \to & \color{green}{(E)} & & & \} \end{array}$$

Eine Ableitung von (2+1) * (2+2):

$$E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * \color{red}{Z} \Rightarrow Z * \color{green}{(E)}$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$\begin{aligned} P = \{ & E \to M, \quad E \to E + M, \quad M \to Z, \quad M \to M * Z, \\ & Z \to 1, \quad Z \to 2, \quad Z \to (E) \quad \} \end{aligned}$$

Eine Ableitung von (2+1) * (2+2):

$$E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M)$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$\begin{aligned} P = \{E &\to M, & E &\to E + M, & M &\to Z, & M &\to M * Z, \\ Z &\to 1, & Z &\to 2, & Z &\to (E) & & \} \end{aligned}$$

Eine Ableitung von (2+1) * (2+2):

$$\begin{aligned} &E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\ &\Rightarrow (E) * (E + M) \end{aligned}$$

Beispiel

$$G = (V, \Sigma, P, E) \text{ mit } V = \{E, M, Z\} \text{ und } \Sigma = \{+, *, 1, 2, (,)\} \qquad \text{und}$$

$$\begin{array}{lllllllllllll} P = \{E & \to & M, & E & \to & E + M, & M & \to & Z, & M & \to & M * Z, & \\ Z & \to & 1, & Z & \to & 2, & Z & \to & (E) & & & & \} \end{array}$$

Eine Ableitung von (2+1) * (2+2):

$$\begin{aligned} & E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\ & \Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \end{aligned}$$

Beispiel

$$G = (V, \Sigma, P, E) \text{ mit } V = \{E, M, Z\} \text{ und } \Sigma = \{+, *, 1, 2, (,)\} \qquad \text{und}$$

$$\begin{array}{llllllllllll} P = \{E & \to & M, & E & \to & E + M, & M & \to & Z, & M & \to & M * Z, \\ Z & \to & 1, & Z & \to & 2, & Z & \to & (E) & & & \} \end{array}$$

Eine Ableitung von (2+1) * (2+2):

$$\begin{aligned} & E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\ & \Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z) \end{aligned}$$

Beispiel

$$G = (V, \Sigma, P, E) \text{ mit } V = \{E, M, Z\} \text{ und } \Sigma = \{+, *, 1, 2, (,)\} \qquad \text{und}$$

$$\begin{aligned} P = \{ E &\rightarrow M, & E &\rightarrow E + M, & M &\rightarrow Z, & M &\rightarrow M * Z, \\ Z &\rightarrow 1, & Z &\rightarrow 2, & Z &\rightarrow (E) & & \quad \} \end{aligned}$$

Eine Ableitung von (2+1) * (2+2):

$$\begin{aligned} & E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\ & \Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z) \\ & \Rightarrow (M + M) * (E + Z) \end{aligned}$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$\begin{aligned} P = \{ & E \to M, && E \to E + M, && M \to Z, && M \to M * Z, \\ & Z \to 1, && Z \to 2, && Z \to (E) && \} \end{aligned}$$

Eine Ableitung von (2+1) * (2+2):

$$\begin{aligned} & E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\ & \Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z) \\ & \Rightarrow (M + M) * (E + Z) \Rightarrow (M + M) * (M + Z) \end{aligned}$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$
\begin{array}{llllllllllll}
P = \{E & \to & M, & E & \to & E + M, & M & \to & Z, & M & \to & M * Z, \\
Z & \to & 1, & Z & \to & 2, & Z & \to & (E) & & & \}
\end{array}
$$

Eine Ableitung von (2+1) * (2+2):

$$
\begin{aligned}
& E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\
& \Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z) \\
& \Rightarrow (M + M) * (E + Z) \Rightarrow (M + M) * (M + Z) \\
& \Rightarrow (M + M) * (Z + Z)
\end{aligned}
$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$
\begin{array}{llllllllllll}
P = \{E & \to & M, & E & \to & E + M, & M & \to & Z, & M & \to & M * Z, \\
Z & \to & 1, & Z & \to & 2, & Z & \to & (E) & & & \}
\end{array}
$$

Eine Ableitung von (2+1) * (2+2):

$$
\begin{aligned}
& E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\
& \Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z) \\
& \Rightarrow (M + M) * (E + Z) \Rightarrow (M + M) * (M + Z) \\
& \Rightarrow (M + M) * (Z + Z) \Rightarrow (M + M) * (Z + 2)
\end{aligned}
$$

Beispiel

$$G = (V, \Sigma, P, E) \text{ mit } V = \{E, M, Z\} \text{ und } \Sigma = \{+, *, 1, 2, (,)\} \qquad \text{und}$$

$$\begin{array}{llllllllllll} P = \{E & \to & M, & E & \to & E + M, & M & \to & Z, & M & \to & M * Z, \\ Z & \to & 1, & Z & \to & 2, & Z & \to & (E) & & & \} \end{array}$$

Eine Ableitung von (2+1) * (2+2):

$$\begin{aligned} & E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\ & \Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z) \\ & \Rightarrow (M + M) * (E + Z) \Rightarrow (M + M) * (M + Z) \\ & \Rightarrow (M + M) * (Z + Z) \Rightarrow (M + M) * (Z + 2) \\ & \Rightarrow (M + Z) * (Z + 2) \end{aligned}$$

Beispiel

$$G = (V, \Sigma, P, E) \text{ mit } V = \{E, M, Z\} \text{ und } \Sigma = \{+, *, 1, 2, (,)\} \qquad \text{und}$$

$$\begin{array}{rlllllllllll} P = \{E & \to & M, & E & \to & E+M, & M & \to & Z, & M & \to & M*Z, \\ Z & \to & 1, & {\color{red}Z} & \to & {\color{green}2}, & Z & \to & (E) & & & \} \end{array}$$

Eine Ableitung von (2+1) * (2+2):

$$\begin{aligned} & E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E+M) \\ & \Rightarrow (E) * (E+M) \Rightarrow (E) * (E+Z) \Rightarrow (E+M) * (E+Z) \\ & \Rightarrow (M+M) * (E+Z) \Rightarrow (M+M) * (M+Z) \\ & \Rightarrow (M+M) * (Z+Z) \Rightarrow (M+M) * (Z+2) \\ & \Rightarrow (M+Z) * ({\color{red}Z}+2) \Rightarrow (M+Z) * ({\color{green}2}+2) \end{aligned}$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$
\begin{aligned}
P = \{ & E \to M, \quad E \to E + M, \quad {\color{red}M} \to {\color{green}Z}, \quad M \to M * Z, \\
& Z \to 1, \quad Z \to 2, \quad Z \to (E) \}
\end{aligned}
$$

Eine Ableitung von (2+1) * (2+2):

$$
\begin{aligned}
& E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\
& \Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z) \\
& \Rightarrow (M + M) * (E + Z) \Rightarrow (M + M) * (M + Z) \\
& \Rightarrow (M + M) * (Z + Z) \Rightarrow (M + M) * (Z + 2) \\
& \Rightarrow (M + Z) * (Z + 2) \Rightarrow ({\color{red}M} + Z) * (2 + 2) \\
& \Rightarrow ({\color{green}Z} + Z) * (2 + 2)
\end{aligned}
$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$
\begin{aligned}
P = \{E &\to M, & E &\to E + M, & M &\to Z, & M &\to M * Z, \\
Z &\to 1, & Z &\to 2, & Z &\to (E) & & \}
\end{aligned}
$$

Eine Ableitung von (2+1) * (2+2):

$$
\begin{aligned}
&E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\
&\Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z) \\
&\Rightarrow (M + M) * (E + Z) \Rightarrow (M + M) * (M + Z) \\
&\Rightarrow (M + M) * (Z + Z) \Rightarrow (M + M) * (Z + 2) \\
&\Rightarrow (M + Z) * (Z + 2) \Rightarrow (M + Z) * (2 + 2) \\
&\Rightarrow (Z + Z) * (2 + 2) \Rightarrow (2 + Z) * (2 + 2)
\end{aligned}
$$

Beispiel

$G = (V, \Sigma, P, E)$ mit $V = \{E, M, Z\}$ und $\Sigma = \{+, *, 1, 2, (,)\}$ und

$$
\begin{array}{llllllllllll}
P = \{E & \to & M, & E & \to & E + M, & M & \to & Z, & M & \to & M * Z, \\
Z & \to & 1, & Z & \to & 2, & Z & \to & (E) & & & \}
\end{array}
$$

Eine Ableitung von (2+1) * (2+2):

$$
\begin{aligned}
& E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M) \\
& \Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z) \\
& \Rightarrow (M + M) * (E + Z) \Rightarrow (M + M) * (M + Z) \\
& \Rightarrow (M + M) * (Z + Z) \Rightarrow (M + M) * (Z + 2) \\
& \Rightarrow (M + Z) * (Z + 2) \Rightarrow (M + Z) * (2 + 2) \\
& \Rightarrow (Z + Z) * (2 + 2) \Rightarrow (2 + Z) * (2 + 2) \\
& \Rightarrow (2 + 1) * (2 + 2)
\end{aligned}
$$

Beispiel: Ableitungen sind nicht eindeutig

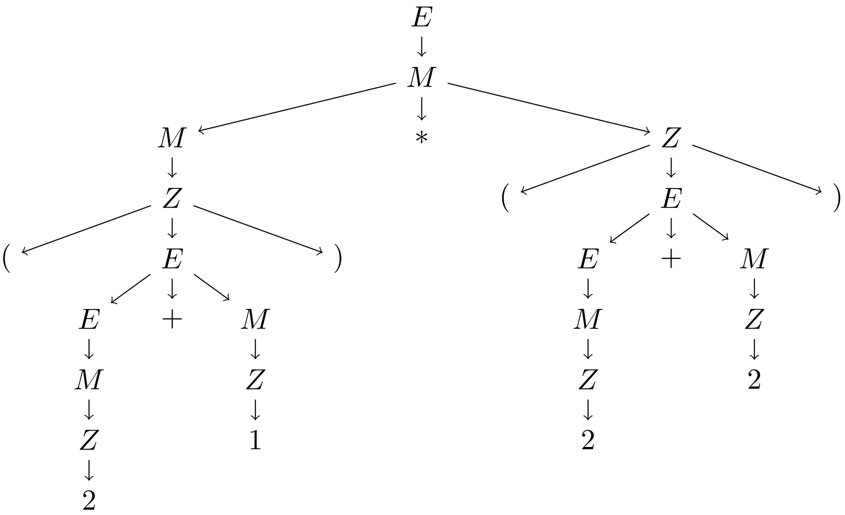
Ableitung von letzter Folie (keine Linksableitung):

$$E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow Z * (E) \Rightarrow Z * (E + M)$$
$$\Rightarrow (E) * (E + M) \Rightarrow (E) * (E + Z) \Rightarrow (E + M) * (E + Z)$$
$$\Rightarrow (M + M) * (E + Z) \Rightarrow (M + M) * (M + Z)$$
$$\Rightarrow (M + M) * (Z + Z) \Rightarrow (M + M) * (Z + 2)$$
$$\Rightarrow (M + Z) * (Z + 2) \Rightarrow (M + Z) * (2 + 2)$$
$$\Rightarrow (Z + Z) * (2 + 2) \Rightarrow (2 + Z) * (2 + 2)$$
$$\Rightarrow (2 + 1) * (2 + 2)$$

Linksableitung: ersetzt immer das linkeste Nichtterminal

$$E \Rightarrow M \Rightarrow M * Z \Rightarrow Z * Z \Rightarrow (E) * Z$$
$$\Rightarrow (E + M) * Z \Rightarrow (M + M) * Z \Rightarrow (Z + M) * Z$$
$$\Rightarrow (2 + M) * Z \Rightarrow (2 + Z) * Z \Rightarrow (2 + 1) * Z \Rightarrow (2 + 1) * (E)$$
$$\Rightarrow (2 + 1) * (E + M) \Rightarrow (2 + 1) * (M + M) \Rightarrow (2 + 1) * (Z + M)$$
$$\Rightarrow (2 + 1) * (2 + M) \Rightarrow (2 + 1) * (2 + Z)$$
$$\Rightarrow (2 + 1) * (2 + 2)$$

Syntaxbaum (zu beiden Ableitungen)



Nichtdeterminismus beim Ableiten

Für eine Satzform u kann es verschiedene Satzformen v_i geben mit $u \Rightarrow_G v_i$.

Quellen des Nichtdeterminismus:

- Wähle, **welche Produktion** $\ell \to r$ aus P angewendet wird
- Wähle die **Position des Teilworts** ℓ in u, das durch r ersetzt wird.

Aber: Es gibt **nur endliche viele** v_i für jeden Schritt!

Erzeugte Sprache

Erzeugte Sprache einer Grammatik

Die von einer Grammatik $G = (V, \Sigma, P, S)$ **erzeugte Sprache** $L(G)$ ist

$$L(G) := \{w \in \Sigma^* \mid S \Rightarrow_G^* w\}.$$

Beispiele

$G_1 = (\{S\}, \{a\}, \{S \to aS\}, S)$
$L(G_1) = \textbf{?}$

$G_2 = (\{S'\}, \{a, b\}, \{S' \to aS', S' \to b\}, S')$
$L(G_2) = \textbf{?}$

Beispiele

$G_1 = (\{S\}, \{a\}, \{S \to aS\}, S)$
$L(G_1) =$ **?**

- $S \Rightarrow aS \Rightarrow aaS \Rightarrow \ldots$ endet nie
- Andere Ableitungen gibt es nicht
- Daher sind keine Worte aus $\{a\}^*$ ableitbar

$G_2 = (\{S'\}, \{a, b\}, \{S' \to aS', S' \to b\}, S')$
$L(G_2) =$ **?**

Beispiele

$G_1 = (\{S\}, \{a\}, \{S \to aS\}, S)$
$L(G_1) = \emptyset$

- $S \Rightarrow aS \Rightarrow aaS \Rightarrow \ldots$ endet nie
- Andere Ableitungen gibt es nicht
- Daher sind keine Worte aus $\{a\}^*$ ableitbar

$G_2 = (\{S'\}, \{a, b\}, \{S' \to aS', S' \to b\}, S')$
$L(G_2) = \textbf{?}$

Beispiele

$G_1 = (\{S\}, \{a\}, \{S \to aS\}, S)$
$L(G_1) = \emptyset$

- $S \Rightarrow aS \Rightarrow aaS \Rightarrow \ldots$ endet nie
- Andere Ableitungen gibt es nicht
- Daher sind keine Worte aus $\{a\}^*$ ableitbar

$G_2 = (\{S'\}, \{a, b\}, \{S' \to aS', S' \to b\}, S')$
$L(G_2) =$ **?**

- $$\begin{tikzcd} S' \arrow[r, Rightarrow] \arrow[d, Rightarrow] & aS' \arrow[r, Rightarrow] \arrow[d, Rightarrow] & aaS' \arrow[r, Rightarrow] \arrow[d, Rightarrow] & aaaS' \arrow[r, Rightarrow] \arrow[d, Rightarrow] & aaaaS' \arrow[r, Rightarrow] \arrow[d, Rightarrow] & \cdots \\ b & ab & aab & aaab & aaaab & \end{tikzcd}$$
- Für alle $i \in \mathbb{N}$ gilt $S \Rightarrow^i a^i S \Rightarrow a^i b$

Beispiele

$G_1 = (\{S\}, \{a\}, \{S \to aS\}, S)$
$L(G_1) = \emptyset$

- $S \Rightarrow aS \Rightarrow aaS \Rightarrow \ldots$ endet nie
- Andere Ableitungen gibt es nicht
- Daher sind keine Worte aus $\{a\}^*$ ableitbar

$G_2 = (\{S'\}, \{a, b\}, \{S' \to aS', S' \to b\}, S')$
$L(G_2) = \{a^n b \mid n \in \mathbb{N}\}$

- $$\begin{array}{ccccccccccc} S' & \Longrightarrow & aS' & \Longrightarrow & aaS' & \Longrightarrow & aaaS' & \Longrightarrow & aaaaS' & \Longrightarrow & \cdots \\ \Downarrow & & \Downarrow & & \Downarrow & & \Downarrow & & \Downarrow & & \\ b & & ab & & aab & & aaab & & aaaab & & \end{array}$$
- Für alle $i \in \mathbb{N}$ gilt $S \Rightarrow^i a^i S \Rightarrow a^i b$

Die Chomsky-Hierarchie

Noam Chomsky teilte die Grammatiken in Typen 0 bis 3:

Sei $G = (V, \Sigma, P, S)$ eine Grammatik.

Die Chomsky-Hierarchie

Noam Chomsky teilte die Grammatiken in Typen 0 bis 3:

Sei $G = (V, \Sigma, P, S)$ eine Grammatik.

G ist vom Typ 0

G ist automatisch vom Typ 0.

G ist vom Typ 1 (kontextsensitive Grammatik), wenn ...

für alle $(\ell \to r) \in P$: $|\ell| \leq |r|$.

G ist vom Typ 2 (kontextfreie Grammatik), wenn ...

G ist vom Typ 1 und für alle $(\ell \to r) \in P$ gilt: $\ell = A \in V$

G ist vom Typ 3 (reguläre Grammatik), wenn ...

G ist vom Typ 2 und für alle $(A \to r) \in P$ gilt: $r = a$ oder $r = aA'$ für $a \in \Sigma, A' \in V$ (die rechten Seiten sind Worte aus $(\Sigma \cup (\Sigma V))$)

Die Chomsky-Hierarchie

Noam Chomsky teilte die Grammatiken in Typen 0 bis 3:

Sei $G = (V, \Sigma, P, S)$ eine Grammatik.

G ist vom Typ 0

G ist automatisch vom Typ 0.

G ist vom Typ 1 (kontextsensitive Grammatik), wenn ...

für alle $(\ell \to r) \in P$: $|\ell| \leq |r|$.

G ist vom Typ 2 (kontextfreie Grammatik), wenn ...

G ist vom Typ 1 und für alle $(\ell \to r) \in P$ gilt: $\ell = A \in V$

G ist vom Typ 3 (reguläre Grammatik), wenn ...

G ist vom Typ 2 und für alle $(A \to r) \in P$ gilt: $r = a$ oder $r = aA'$ für $a \in \Sigma, A' \in V$
(die rechten Seiten sind Worte aus $(\Sigma \cup (\Sigma V))$)

Die Chomsky-Hierarchie

Noam Chomsky teilte die Grammatiken in Typen 0 bis 3:

Sei $G = (V, \Sigma, P, S)$ eine Grammatik.

G ist vom Typ 0

G ist automatisch vom Typ 0.

G ist vom Typ 1 (kontextsensitive Grammatik), wenn ...

für alle $(\ell \to r) \in P$: $|\ell| \leq |r|$.

G ist vom Typ 2 (kontextfreie Grammatik), wenn ...

G ist vom Typ 1 und für alle $(\ell \to r) \in P$ gilt: $\ell = A \in V$

G ist vom Typ 3 (reguläre Grammatik), wenn ...

G ist vom Typ 2 und für alle $(A \to r) \in P$ gilt: $r = a$ oder $r = aA'$ für $a \in \Sigma, A' \in V$
(die rechten Seiten sind Worte aus $(\Sigma \cup (\Sigma V))$)

Die Chomsky-Hierarchie

Noam Chomsky teilte die Grammatiken in Typen 0 bis 3:

Sei $G = (V, \Sigma, P, S)$ eine Grammatik.

G ist vom Typ 0

G ist automatisch vom Typ 0.

G ist vom Typ 1 (kontextsensitive Grammatik), wenn ...

für alle $(\ell \to r) \in P$: $|\ell| \leq |r|$.

G ist vom Typ 2 (kontextfreie Grammatik), wenn ...

G ist vom Typ 1 und für alle $(\ell \to r) \in P$ gilt: $\ell = A \in V$

G ist vom Typ 3 (reguläre Grammatik), wenn ...

G ist vom Typ 2 und für alle $(A \to r) \in P$ gilt: $r = a$ oder $r = aA'$ für $a \in \Sigma, A' \in V$
(die rechten Seiten sind Worte aus $(\Sigma \cup (\Sigma V))$)

Typ i-Sprachen

Definition

Für $i = 0, 1, 2, 3$ nennt man eine formale **Sprache** $L \subseteq \Sigma^*$ **vom Typ** i, falls es eine Typ i-Grammatik G gibt, sodass $L(G) = L$ gilt.

Hierbei wird stets der Typ eindeutig festgelegt, sodass der größtmögliche Grammatik-Typ verwendet wird.

Beispiele

- $G_1 = (\{S\}, \{a, b\}, \{S \to aS, S \to b\}, S)$ ist regulär (Typ 3)
- $G_2 = (\{E, M, Z\}, \{+, *, 1, 2, (,)\}, P, E)$ mit
  $P = \{E \to M,\ \ E \to E + M,\ \ M \to Z,\ \ M \to M * Z,$
  $\quad Z \to 1,\ \ Z \to 2,\ \ Z \to (E)\}$ ist kontextfrei (Typ 2)
- $G_3 = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
  $P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab,$
  $\quad bB \to bb, bC \to bc, cC \to cc\}$ ist kontextsensitiv (Typ 1)

  Beachte $L(G_3) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$
- $G_4 = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit
  $P = \{S \to \$T\$, T \to aAT, T \to bBT, T \to \varepsilon, \$a \to a\$,$
  $\quad \$b \to b\$, Aa \to aA, Ab \to bA, Ba \to aB, Bb \to bB,$
  $\quad A\$ \to \$a, B\$ \to \$b, \$\$ \to \varepsilon\}$ ist vom Typ 0

Beispiele

- $G_1 = (\{S\}, \{a, b\}, \{S \to aS, S \to b\}, S)$ ist regulär (Typ 3)
- $G_2 = (\{E, M, Z\}, \{+, *, 1, 2, (,)\}, P, E)$ mit
  $P = \{E \to M,\ \ E \to E + M,\ \ M \to Z,\ \ M \to M * Z,$
  $Z \to 1,\ \ Z \to 2,\ \ Z \to (E)\}$ ist kontextfrei (Typ 2)
- $G_3 = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
  $P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab,$
  $bB \to bb, bC \to bc, cC \to cc\}$ ist kontextsensitiv (Typ 1)

  Beachte $L(G_3) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$
- $G_4 = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit
  $P = \{S \to \$T\$, T \to aAT, T \to bBT, T \to \varepsilon, \$a \to a\$,$
  $\$b \to b\$, Aa \to aA, Ab \to bA, Ba \to aB, Bb \to bB,$
  $A\$ \to \$a, B\$ \to \$b, \$\$ \to \varepsilon\}$ ist vom Typ 0

Beispiele

- $G_1 = (\{S\}, \{a, b\}, \{S \to aS, S \to b\}, S)$ ist regulär (Typ 3)
- $G_2 = (\{E, M, Z\}, \{+, *, 1, 2, (,)\}, P, E)$ mit
  $P = \{E \to M,\ \ E \to E + M,\ \ M \to Z,\ \ M \to M * Z,$
  $Z \to 1,\ \ Z \to 2,\ \ Z \to (E)\}$ ist kontextfrei (Typ 2)
- $G_3 = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
  $P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab,$
  $bB \to bb, bC \to bc, cC \to cc\}$ ist kontextsensitiv (Typ 1)

  Beachte $L(G_3) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$
- $G_4 = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit
  $P = \{S \to \$T\$, T \to aAT, T \to bBT, T \to \varepsilon, \$a \to a\$,$
  $\$b \to b\$, Aa \to aA, Ab \to bA, Ba \to aB, Bb \to bB,$
  $A\$ \to \$a, B\$ \to \$b, \$\$ \to \varepsilon\}$ ist vom Typ 0

Beispiele

- $G_1 = (\{S\}, \{a, b\}, \{S \to aS, S \to b\}, S)$ ist regulär (Typ 3)
- $G_2 = (\{E, M, Z\}, \{+, *, 1, 2, (,)\}, P, E)$ mit
  $P = \{E \to M,\ \ E \to E + M,\ \ M \to Z,\ \ M \to M * Z,$
  $Z \to 1,\ \ Z \to 2,\ \ Z \to (E)\}$ ist kontextfrei (Typ 2)
- $G_3 = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
  $P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab,$
  $bB \to bb, bC \to bc, cC \to cc\}$ ist kontextsensitiv (Typ 1)

  Beachte $L(G_3) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$
- $G_4 = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit
  $P = \{S \to \$T\$, T \to aAT, T \to bBT, T \to \varepsilon, \$a \to a\$,$
  $\$b \to b\$, Aa \to aA, Ab \to bA, Ba \to aB, Bb \to bB,$
  $A\$ \to \$a, B\$ \to \$b, \$\$ \to \varepsilon\}$ ist vom Typ 0

Erzeugte Sprachen, Mehrdeutige Grammatiken und Sprachen, Entfernen von ε-Produktionen

Prof. Dr. David Sabel

LFE Theoretische Informatik

Letzte Änderung der Folien: 26. April 2022

Wiederholung: Die Chomsky-Hierarchie

Sei $G = (V, \Sigma, P, S)$ eine Grammatik.

G ist vom Typ 0

G ist automatisch vom Typ 0.

G ist vom Typ 1 (kontextsensitive Grammatik), wenn ...

für alle $(\ell \to r) \in P$: $|\ell| \leq |r|$.

G ist vom Typ 2 (kontextfreie Grammatik), wenn ...

G ist vom Typ 1 und für alle $(\ell \to r) \in P$ gilt: $\ell = A \in V$

G ist vom Typ 3 (reguläre Grammatik), wenn ...

G ist vom Typ 2 und für alle $(A \to r) \in P$ gilt: $r = a$ oder $r = aA'$ für $a \in \Sigma, A' \in V$ (die rechten Seiten sind Worte aus $(\Sigma \cup (\Sigma V))$)

Beispiel (kontextsensitive Grammatik)

$G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit

$$P = \{S \to aSBC,\ S \to aBC,\ CB \to BC,\ aB \to ab,\ bB \to bb,\ bC \to bc,\ cC \to cc\}$$

Beispiel-Ableitung:

$$\begin{aligned} S &\Rightarrow aSBC \Rightarrow aaSBCBC \Rightarrow aaaSBCBCBC \Rightarrow aaaaBCBCBCBC \\ &\Rightarrow aaaabCBCBCBC \Rightarrow aaaabBCCBCBC \Rightarrow aaaabbCCBCBC \\ &\Rightarrow aaaabbCBCCBC \Rightarrow aaaabbBCCCBC \Rightarrow aaaabbBCCBCC \\ &\Rightarrow aaaabbBCBCCC \Rightarrow aaaabbBBCCCC \Rightarrow aaaabbbBCCCC \\ &\Rightarrow aaaabbbbCCCC \Rightarrow aaaabbbbcCCC \Rightarrow aaaabbbbccCC \\ &\Rightarrow aaaabbbbcccC \Rightarrow aaaabbbbcccc \end{aligned}$$

Steckengebliebene Folge von Ableitungsschritten:

$$S \Rightarrow aSBC \Rightarrow aaBCBC \Rightarrow aabCBC \Rightarrow aabcBC$$

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\supseteq$“: Zeige $a^n b^n c^n \in L(G)$ für alle $n \in \mathbb{N}_{>0}$

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\supseteq$": Zeige $a^n b^n c^n \in L(G)$ für alle $n \in \mathbb{N}_{>0}$

- Wende $n-1$ mal $S \to aSBC$ und dann einmal $S \to aBC$ an:
  $S \Rightarrow^* a^{n-1}S(BC)^{n-1} \Rightarrow a^n(BC)^n$

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\supseteq$“: Zeige $a^n b^n c^n \in L(G)$ für alle $n \in \mathbb{N}_{>0}$

- Wende $n - 1$ mal $S \to aSBC$ und dann einmal $S \to aBC$ an:
  $S \Rightarrow^* a^{n-1} S (BC)^{n-1} \Rightarrow a^n (BC)^n$
- Wende $CB \to BC$ solange an, bis es kein Teilwort CB mehr gibt.
  $a^n (BC)^n \Rightarrow^* a^n B^n C^n$

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\supseteq$": Zeige $a^n b^n c^n \in L(G)$ für alle $n \in \mathbb{N}_{>0}$

- Wende $n-1$ mal $S \to aSBC$ und dann einmal $S \to aBC$ an:
  $S \Rightarrow^* a^{n-1}S(BC)^{n-1} \Rightarrow a^n(BC)^n$
- Wende $CB \to BC$ solange an, bis es kein Teilwort CB mehr gibt.
  $a^n(BC)^n \Rightarrow^* a^n B^n C^n$
- Wende $aB \to ab$ und anschließend $n-1$ mal $bB \to bb$ an.
  $a^n B^n C^n \Rightarrow a^n b B^{n-1} C^n \Rightarrow^* a^n b^n C^n$

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\supseteq$": Zeige $a^n b^n c^n \in L(G)$ für alle $n \in \mathbb{N}_{>0}$

- Wende $n-1$ mal $S \to aSBC$ und dann einmal $S \to aBC$ an:
  $S \Rightarrow^* a^{n-1} S (BC)^{n-1} \Rightarrow a^n (BC)^n$
- Wende $CB \to BC$ solange an, bis es kein Teilwort CB mehr gibt.
  $a^n (BC)^n \Rightarrow^* a^n B^n C^n$
- Wende $aB \to ab$ und anschließend $n-1$ mal $bB \to bb$ an.
  $a^n B^n C^n \Rightarrow a^n b B^{n-1} C^n \Rightarrow^* a^n b^n C^n$
- Wende einmal $bC \to bc$ und anschließend $n-1$ mal $cC \to cc$ an
  $a^n b^n C^n \Rightarrow a^n b^n c C^{n-1} \Rightarrow^* a^n b^n c^n$

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\supseteq$“: Zeige $a^n b^n c^n \in L(G)$ für alle $n \in \mathbb{N}_{>0}$

- Wende $n-1$ mal $S \to aSBC$ und dann einmal $S \to aBC$ an:
  $S \Rightarrow^* a^{n-1} S (BC)^{n-1} \Rightarrow a^n (BC)^n$
- Wende $CB \to BC$ solange an, bis es kein Teilwort CB mehr gibt.
  $a^n (BC)^n \Rightarrow^* a^n B^n C^n$
- Wende $aB \to ab$ und anschließend $n-1$ mal $bB \to bb$ an.
  $a^n B^n C^n \Rightarrow a^n b B^{n-1} C^n \Rightarrow^* a^n b^n C^n$
- Wende einmal $bC \to bc$ und anschließend $n-1$ mal $cC \to cc$ an
  $a^n b^n C^n \Rightarrow a^n b^n c C^{n-1} \Rightarrow^* a^n b^n c^n$

Zusammensetzen aller Ableitungsschritte zeigt $S \Rightarrow^* a^n b^n c^n$.

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ (2)

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\subseteq$": Zeige, dass alle von G erzeugten Worte von der Form $a^n b^n c^n$ sind.

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ (2)

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\subseteq$“: Zeige, dass alle von G erzeugten Worte von der Form $a^n b^n c^n$ sind.

- Für $S \Rightarrow_G^* u$ mit u Satzform zeigen die Regeln:
  $\#_a(u) = \#_b(u) + \#_B(u) = \#_c(u) + \#_C(u)$

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ (2)

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\subseteq$": Zeige, dass alle von G erzeugten Worte von der Form $a^n b^n c^n$ sind.

- Für $S \Rightarrow_G^* u$ mit u Satzform zeigen die Regeln:
  $\#_a(u) = \#_b(u) + \#_B(u) = \#_c(u) + \#_C(u)$
- Für $S \Rightarrow_G^* w$ mit $w \in \{a, b, c\}^*$ gilt: a's werden ganz links erzeugt, d.h. $w = a^n w'$ mit $w' \in \{b, c\}^*$ und $n = \#_b(w') = \#_c(w')$

Grammatik erzeugt $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ (2)

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\subseteq$“: Zeige, dass alle von G erzeugten Worte von der Form $a^n b^n c^n$ sind.

- Für $S \Rightarrow_G^* u$ mit u Satzform zeigen die Regeln:
  $\#_a(u) = \#_b(u) + \#_B(u) = \#_c(u) + \#_C(u)$
- Für $S \Rightarrow_G^* w$ mit $w \in \{a, b, c\}^*$ gilt: a's werden ganz links erzeugt, d.h. $w = a^n w'$ mit $w' \in \{b, c\}^*$ und $n = \#_b(w') = \#_c(w')$
- Es gilt $w' = bw_1$, da jedes auf a folgende Symbol durch $aB \to ab$ erzeugt wird und die Regeln keine Terminalsymbole vertauschen.

Grammatik, die $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ erzeugt (3)

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\subseteq$": Zeige, dass alle von G erzeugten Worte von der Form $a^n b^n c^n$ sind.

- ...
- Ebenso können die Terminalsymbole des Wortes $w' \in \{b, c\}^*$ nur durch $bB \to bb$, $bC \to bc$ und $cC \to cc$ erzeugt worden sein.
  Diese Produktionen erlauben nur einen Wechsel von b zu c und keine Wechsel von c zu b. Auch ein Umordnen der Terminalsymbole ist nicht möglich (da es keine Produktion dafür gibt).
- Daher gilt $w' = b^i c^j$ und mit $n = \#_b(w') = \#_c(w')$ sogar $w' = b^n c^n$. $\square$

Grammatik, die $\{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ erzeugt (3)

Satz

$L(G) = \{a^n b^n c^n \mid n \in \mathbb{N}_{>0}\}$ für $G = (\{S, B, C\}, \{a, b, c\}, P, S)$ mit
$P = \{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc\}$

„$\subseteq$“: Zeige, dass alle von G erzeugten Worte von der Form $a^n b^n c^n$ sind.

- ...
- Ebenso können die Terminalsymbole des Wortes $w' \in \{b, c\}^*$ nur durch $bB \to bb$, $bC \to bc$ und $cC \to cc$ erzeugt worden sein.
  Diese Produktionen erlauben nur einen Wechsel von b zu c und keine Wechsel von c zu b. Auch ein Umordnen der Terminalsymbole ist nicht möglich (da es keine Produktion dafür gibt).
- Daher gilt $w' = b^i c^j$ und mit $n = \#_b(w') = \#_c(w')$ sogar $w' = b^n c^n$. $\square$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,$$
$$Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$S$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ T \to aAT,\ T \to bBT,\ T \to \varepsilon,\ \$a \to a\$,\ \$b \to b\$,\ Aa \to aA,\\ Ab \to bA,\ Ba \to aB,\ Bb \to bB,\ A\$ \to \$a,\ B\$ \to \$b,\ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$S \Rightarrow \$T\$$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$S \Rightarrow \$T\$ \Rightarrow \$aAT\$$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ T \to aAT,\ T \to bBT,\ T \to \varepsilon,\ \$a \to a\$,\ \$b \to b\$,\ Aa \to aA,\\ Ab \to bA,\ Ba \to aB,\ Bb \to bB,\ A\$ \to \$a,\ B\$ \to \$b,\ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$S \Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$S \Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ T \to aAT,\ T \to bBT,\ T \to \varepsilon,\ \$a \to a\$,\ \$b \to b\$,\ Aa \to aA,\ Ab \to bA,\ Ba \to aB,\ Bb \to bB,\ A\$ \to \$a,\ B\$ \to \$b,\ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,$$
$$Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ T \to aAT,\ T \to bBT,\ T \to \varepsilon,\ \$a \to a\$,\ \$b \to b\$,\ Aa \to aA,\ Ab \to bA,\ Ba \to aB,\ Bb \to bB,\ A\$ \to \$a,\ B\$ \to \$b,\ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \Rightarrow \$aabAAB\$ \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \Rightarrow \$aabAAB\$ \\ &\Rightarrow \$aabAA\$b \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \Rightarrow \$aabAAB\$ \\ &\Rightarrow \$aabAA\$b \Rightarrow \$aabA\$ab \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \Rightarrow \$aabAAB\$ \\ &\Rightarrow \$aabAA\$b \Rightarrow \$aabA\$ab \Rightarrow \$aab\$aab \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \Rightarrow \$aabAAB\$ \\ &\Rightarrow \$aabAA\$b \Rightarrow \$aabA\$ab \Rightarrow \$aab\$aab \Rightarrow a\$ab\$aab \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \Rightarrow \$aabAAB\$ \\ &\Rightarrow \$aabAA\$b \Rightarrow \$aabA\$ab \Rightarrow \$aab\$aab \Rightarrow a\$ab\$aab \\ &\Rightarrow aa\$b\$aab \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \Rightarrow \$aabAAB\$ \\ &\Rightarrow \$aabAA\$b \Rightarrow \$aabA\$ab \Rightarrow \$aab\$aab \Rightarrow a\$ab\$aab \\ &\Rightarrow aa\$b\$aab \Rightarrow aab\$\$aab \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ T \to aAT,\ T \to bBT,\ T \to \varepsilon,\ \$a \to a\$,\ \$b \to b\$,\ Aa \to aA,\ Ab \to bA,\ Ba \to aB,\ Bb \to bB,\ A\$ \to \$a,\ B\$ \to \$b,\ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \Rightarrow \$aabAAB\$ \\ &\Rightarrow \$aabAA\$b \Rightarrow \$aabA\$ab \Rightarrow \$aab\$aab \Rightarrow a\$ab\$aab \\ &\Rightarrow aa\$b\$aab \Rightarrow aab\$\$aab \Rightarrow aabaab \end{aligned}$$

Beispiel einer Typ 0-Grammatik

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Eine Ableitung:

$$\begin{aligned} S &\Rightarrow \$T\$ \Rightarrow \$aAT\$ \Rightarrow \$aAaAT\$ \Rightarrow \$aAaAbBT\$ \\ &\Rightarrow \$aAaAbB\$ \Rightarrow \$aaAAbB\$ \Rightarrow \$aaAbAB\$ \Rightarrow \$aabAAB\$ \\ &\Rightarrow \$aabAA\$b \Rightarrow \$aabA\$ab \Rightarrow \$aab\$aab \Rightarrow a\$ab\$aab \\ &\Rightarrow aa\$b\$aab \Rightarrow aab\$\$aab \Rightarrow aabaab \end{aligned}$$

Beachte: $L(G) = \{ww \mid w \in \{a, b\}^*\}$ und $L(G)$ ist Typ 1-Sprache

Beispiel einer Typ 0-Grammatik (2)

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Begründung dafür, dass $L(G) = \{ww \mid w \in \{a, b\}^*\}$ gilt:

Beispiel einer Typ 0-Grammatik (2)

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ T \to aAT,\ T \to bBT,\ T \to \varepsilon,\ \$a \to a\$,\ \$b \to b\$,\ Aa \to aA,\\ Ab \to bA,\ Ba \to aB,\ Bb \to bB,\ A\$ \to \$a,\ B\$ \to \$b,\ \$\$ \to \varepsilon\}$$

Begründung dafür, dass $L(G) = \{ww \mid w \in \{a, b\}^*\}$ gilt:

- Mit $S \to \$T\$$ wird zunächst eine Umrahmung mit $\$\$$ erzeugt

Beispiel einer Typ 0-Grammatik (2)

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Begründung dafür, dass $L(G) = \{ww \mid w \in \{a, b\}^*\}$ gilt:

- Mit $S \to \$T\$$ wird zunächst eine Umrahmung mit $\$\$$ erzeugt
- Mit $T \to aAT$, $T \to bBT$, $T \to \varepsilon$ wird ein Wort aus 2er Blöcken aA und bB erzeugt

Beispiel einer Typ 0-Grammatik (2)

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Begründung dafür, dass $L(G) = \{ww \mid w \in \{a, b\}^*\}$ gilt:

- Mit $S \to \$T\$$ wird zunächst eine Umrahmung mit $\$\$$ erzeugt
- Mit $T \to aAT$, $T \to bBT$, $T \to \varepsilon$ wird ein Wort aus 2er Blöcken aA und bB erzeugt
- Mit $Aa \to aA, Ab \to bA, Ba \to aB, Bb \to bB$ werden A's und B's bis vor $\$$ geschoben

Beispiel einer Typ 0-Grammatik (2)

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,\\ Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Begründung dafür, dass $L(G) = \{ww \mid w \in \{a, b\}^*\}$ gilt:

- Mit $S \to \$T\$$ wird zunächst eine Umrahmung mit \$\$ erzeugt
- Mit $T \to aAT$, $T \to bBT$, $T \to \varepsilon$ wird ein Wort aus 2er Blöcken aA und bB erzeugt
- Mit $Aa \to aA, Ab \to bA, Ba \to aB, Bb \to bB$ werden A's und B's bis vor \$ geschoben
- Mit $A\$ \to \$a$ und $B\$ \to \$b$ werden die A's und B's in a's und b's verwandelt, indem sie über das rechte \$ hüpfen.

Beispiel einer Typ 0-Grammatik (2)

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,$$
$$Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Begründung dafür, dass $L(G) = \{ww \mid w \in \{a, b\}^*\}$ gilt:

- Mit $S \to \$T\$$ wird zunächst eine Umrahmung mit $\$\$$ erzeugt
- Mit $T \to aAT$, $T \to bBT$, $T \to \varepsilon$ wird ein Wort aus 2er Blöcken aA und bB erzeugt
- Mit $Aa \to aA, Ab \to bA, Ba \to aB, Bb \to bB$ werden A's und B's bis vor $\$$ geschoben
- Mit $A\$ \to \$a$ und $B\$ \to \$b$ werden die A's und B's in a's und b's verwandelt, indem sie über das rechte $\$$ hüpfen.
- Mit $\$a \to a\$$, $\$b \to b\$$ wird das linke $\$$ zum rechten geschoben, mit $\$\$ \to \varepsilon$ werden sie dann eliminiert.

Beispiel einer Typ 0-Grammatik (2)

Grammatik $G = (\{S, T, A, B, \$\}, \{a, b\}, P, S)$ mit

$$P = \{S \to \$T\$,\ \ T \to aAT,\ \ T \to bBT,\ \ T \to \varepsilon,\ \ \$a \to a\$,\ \ \$b \to b\$,\ \ Aa \to aA,$$
$$Ab \to bA,\ \ Ba \to aB,\ \ Bb \to bB,\ \ A\$ \to \$a,\ \ B\$ \to \$b,\ \ \$\$ \to \varepsilon\}$$

Begründung dafür, dass $L(G) = \{ww \mid w \in \{a,b\}^*\}$ gilt:

- Mit $S \to \$T\$$ wird zunächst eine Umrahmung mit $\$\$$ erzeugt
- Mit $T \to aAT$, $T \to bBT$, $T \to \varepsilon$ wird ein Wort aus 2er Blöcken aA und bB erzeugt
- Mit $Aa \to aA, Ab \to bA, Ba \to aB, Bb \to bB$ werden A's und B's bis vor $\$$ geschoben
- Mit $A\$ \to \$a$ und $B\$ \to \$b$ werden die A's und B's in a's und b's verwandelt, indem sie über das rechte $\$$ hüpfen.
- Mit $\$a \to a\$$, $\$b \to b\$$ wird das linke $\$$ zum rechten geschoben, mit $\$\$ \to \varepsilon$ werden sie dann eliminiert.
- Bei allen Schritten wird die relative Lage aller a zu b sowie aller A zu B nicht geändert.

Mehrdeutige Grammatiken

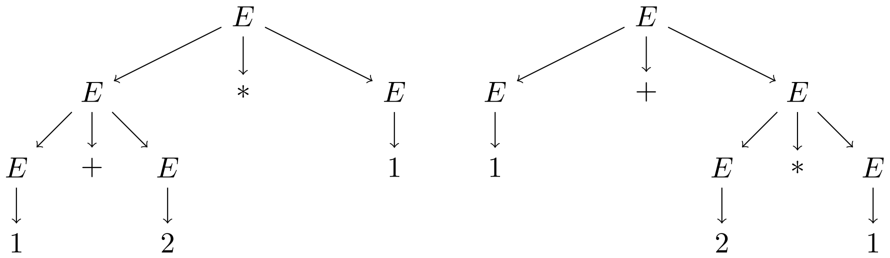
Beispiel:

$$(E, \{*, +, 1, 2\}, \{E \to E * E,\ \ E \to E + E,\ \ E \to 1,\ \ E \to 2\}, E)$$

Zwei Ableitungen für $1 + 2 * 1$:

- $E \Rightarrow E * E \Rightarrow E + E * E \Rightarrow 1 + E * E \Rightarrow 1 + 2 * E \Rightarrow 1 + 2 * 1$
- $E \Rightarrow E + E \Rightarrow E + E * E \Rightarrow 1 + E * E \Rightarrow 1 + 2 * E \Rightarrow 1 + 2 * 1.$

Syntaxbäume dazu:



Mehrdeutige Grammatiken (2)

Mehrdeutige Grammatik

Eine Typ 2-Grammatik ist mehrdeutig, wenn es verschieden strukturierte Syntaxbäume für dasselbe Wort w gibt.

Inhärent mehrdeutige Sprache

Eine Typ 2-Sprache ist inhärent mehrdeutig, wenn es nur mehrdeutige Grammatiken gibt, die diese Sprache erzeugen.

Die Sprache

$$\{a^m b^m c^n d^n \mid m, n \in \mathbb{N}_{>0}\} \cup \{a^m b^n c^n d^m \mid m, n \in \mathbb{N}_{>0}\}$$

ist inhärent mehrdeutig (Beweis z.B. in Hopcroft, Motwani, Ullman, 2006)

ε-Regel für Typ 1,2,3-Grammatiken

- Das leere Wort ε kann bisher nicht für Typ 1,2,3 Grammatiken erzeugt werden:

  Produktion $S \to \varepsilon$ erfüllt die Typ 1-Bedingung $|S| \leq |\varepsilon|$ nicht. Daher Sonderregel:

ε-Regel für Typ 1-Grammatiken

Eine Grammatik $G = (V, \Sigma, P, S)$ vom Typ 1 darf eine Produktion $(S \to \varepsilon) \in P$ enthalten, vorausgesetzt, dass keine rechte Seite einer Produktion in P, die Variable S enthält.

Sonderregel erlaubt nicht:

$$G = (\{S\}, \{a\}, \{S \to \varepsilon, S \to aSa\}, S)$$

Sonderregel erlaubt:

$$G = (\{S', S\}, \{a\}, \{S' \to \varepsilon, S' \to aSa, S' \to aa, S \to aSa, S \to aa\}, S')$$

Leeres Wort hinzufügen geht mit Sonderregel immer

Satz

Sei $G = (V, \Sigma, P, S)$ vom Typ $i \in \{1, 2, 3\}$ mit $\varepsilon \notin L(G)$. Sei $S' \notin V$.
Dann erzeugt $G' = (V \cup \{S'\}, \Sigma, P \cup \{S' \to \varepsilon\} \cup \{S' \to r \mid S \to r \in P\}, S')$ die Sprache $L(G') = L(G) \cup \{\varepsilon\}$ und
G' erfüllt die ε-Regel für Typ 1,2,3-Grammatiken und G' ist vom Typ i.

Leeres Wort hinzufügen geht mit Sonderregel immer

Satz

Sei $G = (V, \Sigma, P, S)$ vom Typ $i \in \{1, 2, 3\}$ mit $\varepsilon \notin L(G)$. Sei $S' \notin V$.
Dann erzeugt $G' = (V \cup \{S'\}, \Sigma, P \cup \{S' \to \varepsilon\} \cup \{S' \to r \mid S \to r \in P\}, S')$ die Sprache $L(G') = L(G) \cup \{\varepsilon\}$ und
G' erfüllt die ε-Regel für Typ 1,2,3-Grammatiken und G' ist vom Typ i.

Beweis:

- Da S' neu, kommt S' auf keiner rechten Seite vor.
- Da $S \to r \in P$ vom Typ i, sind auch $S' \to r$ vom Typ i
- Da $S' \Rightarrow \varepsilon$, gilt $\varepsilon \in L(G')$
- Für $w \neq \varepsilon$ gilt: $S \Rightarrow_G^* w$ g.d.w. $S' \Rightarrow_{G'}^* w$
  Der jeweils erste Ableitungsschritt muss ausgetauscht werden, d.h. $S \Rightarrow_G r$ vs. $S' \Rightarrow_{G'} r$

ε-Produktionen für Typ 2- und Typ 3-Grammatiken

Sonderregel für Typ 2- und Typ 3-Grammatiken:

ε-Produktionen in kontextfreien und regulären Grammatiken

In Grammatiken des Typs 2 und des Typs 3 erlauben wir Produktionen der Form $A \to \varepsilon$ (sogenannte ε-Produktionen).

Das ist keine echte Erweiterung, denn:

Satz (Entfernen von ε-Produktionen)

Sei $G = (V, \Sigma, P, S)$ eine kontextfreie (bzw. reguläre) Grammatik mit $\varepsilon \notin L(G)$. Dann gibt es eine kontextfreie (bzw. reguläre) Grammatik G' mit $L(G) = L(G')$ und G' enthält keine ε-Produktionen.

Beweis: Algorithmus auf der nächsten Folie.

Algorithmus 1: Entfernen von ε-Produktionen

Eingabe: Typ i-Grammatik $G = (V, \Sigma, P, S)$ mit $\varepsilon \notin L(G)$, $i \in \{2, 3\}$
Ausgabe: Typ i-Grammatik G' ohne ε-Produktionen, sodass $L(G) = L(G')$
Beginn
  finde die Menge $W \subseteq V$ aller Variablen A für die gilt $A \Rightarrow^* \varepsilon$:
  Beginn
    $W := \{A \mid (A \to \varepsilon) \in P\}$;
    wiederhole
      füge alle A zu W hinzu mit $A \to A_1 \dots A_n \in P$ und $\forall i : A_i \in W$;
    bis *sich* W *nicht mehr ändert*;
  Ende
  $P' := P \setminus \{A \to \varepsilon \mid (A \to \varepsilon) \in P\}$; /* lösche Regeln $A \to \varepsilon$ */
  wiederhole
    für alle *Produktionen* $A' \to uAv \in P'$ *mit* $|uv| > 0$ *und* $A \in W$ **tue**
      füge die Produktion $A' \to uv$ zu P' hinzu;
      /* für $A' \to u'Av'Aw'$ gibt es (mindestens) zwei Hinzufügungen: Für das Vorkommen von A nach u' als auch für das Vorkommen direkt vor w' */
    Ende
  bis *sich* P' *nicht mehr ändert*;
  gebe $G' = (V, \Sigma, P', S)$ als Ergebnisgrammatik aus;
Ende

Algorithmus 1: Entfernen von ε-Produktionen

Eingabe: Typ i-Grammatik $G = (V, \Sigma, P, S)$ mit $\varepsilon \notin L(G)$, $i \in \{2, 3\}$
Ausgabe: Typ i-Grammatik G' ohne ε-Produktionen, sodass $L(G) = L(G')$
Beginn
  finde die Menge $W \subseteq V$ aller Variablen A für die gilt $A \Rightarrow^* \varepsilon$:
  Beginn
    $W := \{A \mid (A \to \varepsilon) \in P\}$;
    wiederhole
      füge alle A zu W hinzu mit $A \to A_1 \ldots A_n \in$
    bis *sich W nicht mehr ändert*;
  Ende
  $P' := P \setminus \{A \to \varepsilon \mid (A \to \varepsilon) \in P\}$; /* lösche Regeln $A \to \varepsilon$ */
  wiederhole
    für alle *Produktionen $A' \to uAv \in P'$ mit $|uv| > 0$ und $A \in W$* **tue**
      füge die Produktion $A' \to uv$ zu P' hinzu;
      /* für $A' \to u'Av'Aw'$ gibt es (mindestens) zwei Hinzufügungen: Für das Vorkommen von A nach u' als auch für das Vorkommen direkt vor w' */
    Ende
  bis *sich P' nicht mehr ändert*;
  gebe $G' = (V, \Sigma, P', S)$ als Ergebnisgrammatik aus;
Ende

Die neuen Produktionen nehmen den Ableitungsschritt $A \to \varepsilon$ vorweg.

Für reguläre Produktion $A' \to aA$ wird $A' \to a$ hinzugefügt (bleibt regulär!)

Beispiel: Entfernen von ε-Produktionen

$G = (\{A, B, C, D, S\}, \{0, 1\}, P, S)$ mit

$$P = \{S \to 1A,\ \ A \to AB,\ \ A \to DA,\ \ A \to \varepsilon,\ \ B \to 0,\\ B \to 1,\ \ C \to AAA,\ \ D \to 1AC\}.$$

Beispiel: Entfernen von ε-Produktionen

$G = (\{A, B, C, D, S\}, \{0, 1\}, P, S)$ mit

$$P = \{S \to 1A,\ \ A \to AB,\ \ A \to DA,\ \ A \to \varepsilon,\ \ B \to 0,\\ B \to 1,\ \ C \to AAA,\ \ D \to 1AC\}.$$

- Menge W der Variablen, die ε herleiten:

  $W = \{A, C\}$ da $A \to \varepsilon$ und $C \to AAA$

Beispiel: Entfernen von ε-Produktionen

$G = (\{A, B, C, D, S\}, \{0, 1\}, P, S)$ mit

$$P = \{S \to 1A,\ \ A \to AB,\ \ A \to DA,\ \ A \to \varepsilon,\ \ B \to 0,$$
$$B \to 1,\ \ C \to AAA,\ \ D \to 1AC\}.$$

- Menge W der Variablen, die ε herleiten:

  $W = \{A, C\}$ da $A \to \varepsilon$ und $C \to AAA$
- Starte mit

  $P' = \{S \to 1A,\ \ A \to AB,\ \ A \to DA,\ \ B \to 0,$
  $\quad B \to 1,\ \ C \to AAA,\ \ D \to 1AC\}.$

Beispiel: Entfernen von ε-Produktionen

$G = (\{A, B, C, D, S\}, \{0, 1\}, P, S)$ mit

$$P = \{S \to 1A,\ \ A \to AB,\ \ A \to DA,\ \ A \to \varepsilon,\ \ B \to 0,\\ B \to 1,\ \ C \to AAA,\ \ D \to 1AC\}.$$

- Menge W der Variablen, die ε herleiten:

  $W = \{A, C\}$ da $A \to \varepsilon$ und $C \to AAA$
- Starte mit

  $$P' = \{S \to 1A,\ \ A \to AB,\ \ A \to DA,\ \ B \to 0,\\ B \to 1,\ \ C \to AAA,\ \ D \to 1AC\}.$$
- Hinzufügen von Produktionen für Vorkommen von A und C

  $$P' = \{S \to 1A,\ \ S \to 1,\ \ A \to AB,\ \ A \to B,\ \ A \to DA,\ \ A \to D,\\ B \to 0,\ \ B \to 1,\ \ C \to AAA,\ \ C \to AA,\ \ C \to A,\\ D \to 1AC,\ \ D \to 1A,\ \ D \to 1C,\ \ D \to 1\}.$$