Grundbegriffe der Informatik

Kapitel 12: kontextfreie Grammatiken

Thomas Worsch

KIT, Institut für Theoretische Informatik

Wintersemester 2015/2016

Überblick

Rekursive Definition syntaktischer Strukturen

Kontextfreie Grammatiken

Relationen (Teil 2)

Eine Grenze kontextfreier Grammatiken

Wo sind wir?

Rekursive Definition syntaktischer Strukturen

Kontextfreie Grammatiken

Relationen (Teil 2)

Eine Grenze kontextfreier Grammatiken

Spezifikation formaler Sprachen

Beschreibung formaler Sprachen nur mit Hilfe

- einzelner Symbole und
- der Operationen Vereinigung, Konkatenation und Konkatenationsabschluss

Spezifikation formaler Sprachen

Beschreibung formaler Sprachen nur mit Hilfe

- einzelner Symbole und
- der Operationen Vereinigung, Konkatenation und Konkatenationsabschluss

ist

- manchmal möglich
- manchmal nicht (Beweis später)

Ausschnitt der Definition der Syntax von Java

1 Block:
{ BlockStatements$_{\text{opt}}$ }

2 BlockStatements:
BlockStatement
BlockStatements BlockStatement

3 BlockStatement:
Statement
......

4 Statement:
StatementWithoutTrailingSubstatement
......

5 StatementWithoutTrailingSubstatement:
Block
......

siehe: `docs.oracle.com/javase/specs/jls/se7/html/jls-14.html`

Rekursion

Definition von ⟨*BlockStatements*⟩ nimmt Bezug auf ⟨*BlockStatements*⟩

Definition von ⟨*Block*⟩ nimmt (indirekt) Bezug auf ⟨*Statement*⟩

Definition von ⟨*Statement*⟩ nimmt (indirekt) Bezug auf auf ⟨*Block*⟩

Was soll das bedeuten?

Vereinfachungen

1	Block:	
		{ Blockstatements }
2	BlockStatements:	
		BlockStatement
		BlockStatements BlockStatement
3	BlockStatement:	
		Statement
		……
4	Statement:	
		StatementWithoutTrailingSubstatement
		……
5	StatementWithoutTrailingSubstatement:	
		Block
		……

Vereinfachungen

1	X
	(X)
2	BlockStatements:
	BlockStatement
	BlockStatements BlockStatement
3	BlockStatement:
	Statement
	……
4	Statement:
	StatementWithoutTrailingSubstatement
	……
5	StatementWithoutTrailingSubstatement:
	Block
	……

Vereinfachungen

1	X	
		(X)
2	X	
		X
		$X\ X$
3	BlockStatement:	
		Statement
		……
4	Statement:	
		StatementWithoutTrailingSubstatement
		……
5	StatementWithoutTrailingSubstatement:	
		Block
		……

Vereinfachungen

1	X	
		(X)
2	X	
		X
		$X\ X$
3	X	
		X
		ε
4	Statement:	
		StatementWithoutTrailingSubstatement
		……
5	StatementWithoutTrailingSubstatement:	
		Block
		……

Vereinfachungen

1	X	
		$(\,X\,)$
2	X	
		X
		$X\,X$
3	X	
		X
		ε
4	X	
		X
5	StatementWithoutTrailingSubstatement:	
		Block
		……

Vereinfachungen

1	X	
		(X)
2	X	
		X
		$X\,X$
3	X	
		X
		ε
4	X	
		X
5	X	
		X

Vereinfachungen

es bleibt

- K1: X:

  ε
- K2: X:

  XX
- K3: X:

  (X)
- K4: Auch gemeint: nichts anderes „ist ein X“

Versuch einer formalen Sprache

versuche, mit X eine formale Sprache L zu assoziieren:

$$L = \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$$

trügerische Hoffnung:

- die Inklusion $L \supseteq \ldots$ spiegelt K1, K2, K3 wider
- die Inklusion $L \subseteq \ldots$ spiegelt K4 wider

Fragen:

1. Gleichung lösbar?
   wäre schön,
2. Lösung eindeutig?
   wäre schön,

Versuch einer formalen Sprache

versuche, mit X eine formale Sprache L zu assoziieren:

$$L = \{\varepsilon\} \cup LL \cup \{\,(\,\}L\{\,)\,\}$$

trügerische Hoffnung:

- die Inklusion $L \supseteq \ldots$ spiegelt K1, K2, K3 wider
- die Inklusion $L \subseteq \ldots$ spiegelt K4 wider

Fragen:

1. Gleichung lösbar?
   wäre schön, und ja, das ist so
2. Lösung eindeutig?
   wäre schön,

Versuch einer formalen Sprache

versuche, mit X eine formale Sprache L zu assoziieren:

$$L = \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$$

trügerische Hoffnung:

- die Inklusion $L \supseteq \ldots$ spiegelt K1, K2, K3 wider
- die Inklusion $L \subseteq \ldots$ spiegelt K4 wider

Fragen:

1. Gleichung lösbar?
   wäre schön, und ja, das ist so
2. Lösung eindeutig?
   wäre schön, aber nein, das ist nicht so

Versuch einer formalen Sprache

versuche, mit X eine formale Sprache L zu assoziieren:

$$L = \{\varepsilon\} \cup LL \cup \{\,(\,\}L\{\,)\,\}$$

trügerische Hoffnung:

- die Inklusion $L \supseteq \ldots$ spiegelt K1, K2, K3 wider
- die Inklusion $L \subseteq \ldots$ spiegelt K4 wider

Fragen:

1. Gleichung lösbar?
   wäre schön, und ja, das ist so
2. Lösung eindeutig?
   wäre schön, aber nein, das ist nicht so

   $\Longrightarrow$ finde und charakterisiere die «interessierende» Lösung

Lösbarkeit von $L = \{\varepsilon\} \cup LL \cup \{\,(\,\}L\{\,)\,\}$

konstruiere Folge $L_0, L_1, \ldots$ formaler Sprachen L_i

- $L_0 = \{\varepsilon\}$.
- für $i \in \mathbb{N}_0$ sei $L_{i+1} = L_i L_i \cup \{\,(\,\}L_i\{\,)\,\}$

Lemma. $L = \bigcup_{i \in \mathbb{N}_0} L_i$ erfüllt die Gleichung.

Beweisstruktur: zeige

- $L \subseteq \{\varepsilon\} \cup LL \cup \{\,(\,\}L\{\,)\,\}$
- $L \supseteq \{\varepsilon\} \cup LL \cup \{\,(\,\}L\{\,)\,\}$

Beweis des Lemmas — Teil 1

für jedes $i \in \mathbb{N}_0$: $\varepsilon \in L_i$, denn:

- $\varepsilon \in L_0$
- für alle $i \in \mathbb{N}_0$ ist $L_i L_i \subseteq L_{i+1}$,
  wenn $\varepsilon \in L_i$, dann auch $\varepsilon = \varepsilon\varepsilon \in L_i L_i \subseteq L_{i+1}$.

Beweis des Lemmas — Teil 1

für jedes $i \in \mathbb{N}_0$: $\varepsilon \in L_i$, denn:

- $\varepsilon \in L_0$
- für alle $i \in \mathbb{N}_0$ ist $L_i L_i \subseteq L_{i+1}$,
  wenn $\varepsilon \in L_i$, dann auch $\varepsilon = \varepsilon\varepsilon \in L_i L_i \subseteq L_{i+1}$.

also für jedes $i \in \mathbb{N}_0$: $L_i \subseteq L_{i+1}$, denn
$L_i = L_i\{\varepsilon\} \subseteq L_i L_i \subseteq L_{i+1}$.

Beweis des Lemmas — Teil 1

für jedes $i \in \mathbb{N}_0$: $\varepsilon \in L_i$, denn:

- $\varepsilon \in L_0$
- für alle $i \in \mathbb{N}_0$ ist $L_i L_i \subseteq L_{i+1}$,
  wenn $\varepsilon \in L_i$, dann auch $\varepsilon = \varepsilon\varepsilon \in L_i L_i \subseteq L_{i+1}$.

also für jedes $i \in \mathbb{N}_0$: $L_i \subseteq L_{i+1}$, denn
$L_i = L_i\{\varepsilon\} \subseteq L_i L_i \subseteq L_{i+1}$.

Zeige: $L \subseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$

- Da $\varepsilon \in L_0 \subseteq L$ ist, ist $L = L\{\varepsilon\} \subseteq LL \subseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$.

Beweis des Lemmas — Teil 1

für jedes $i \in \mathbb{N}_0$: $\boxed{\varepsilon \in L_i}$, denn:

- $\varepsilon \in L_0$
- für alle $i \in \mathbb{N}_0$ ist $L_i L_i \subseteq L_{i+1}$,
  wenn $\varepsilon \in L_i$, dann auch $\varepsilon = \varepsilon\varepsilon \in L_i L_i \subseteq L_{i+1}$.

also für jedes $i \in \mathbb{N}_0$: $\boxed{L_i \subseteq L_{i+1}}$, denn
$L_i = L_i\{\varepsilon\} \subseteq L_i L_i \subseteq L_{i+1}$.

Zeige: $\boxed{L \subseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}}$

- Da $\varepsilon \in L_0 \subseteq L$ ist, ist $L = L\{\varepsilon\} \subseteq LL \subseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$.

Zeige: $\boxed{L \supseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}}$

- sei $w \in \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$.

Beweis des Lemmas — Teil 2

wir wissen

- für jedes $i \in \mathbb{N}_0$: $\varepsilon \in L_i$
- für jedes $i \in \mathbb{N}_0$: $L_i \subseteq L_{i+1}$
- $L \subseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$

noch zu zeigen: $L \supseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$

- sei $w \in \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$.

Beweis des Lemmas — Teil 2

wir wissen

- für jedes $i \in \mathbb{N}_0$: $\varepsilon \in L_i$
- für jedes $i \in \mathbb{N}_0$: $L_i \subseteq L_{i+1}$
- $L \subseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$

noch zu zeigen: $L \supseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$

- sei $w \in \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$.
- 1. Fall: $w = \varepsilon \in L_0 \subseteq L$.

Beweis des Lemmas — Teil 2

wir wissen

- für jedes $i \in \mathbb{N}_0$: $\varepsilon \in L_i$
- für jedes $i \in \mathbb{N}_0$: $L_i \subseteq L_{i+1}$
- $L \subseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$

noch zu zeigen: $L \supseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$

- sei $w \in \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$.
- 1. Fall: $w = \varepsilon \in L_0 \subseteq L$.
- 2. Fall: $w \in LL$:
  - $w = w_1 w_2$ mit $w_1 \in L$ und $w_2 \in L$
  - $w_1 \in L_{i_1}$ und $w_2 \in L_{i_2}$
  - $w_1 \in L_i$ und $w_2 \in L_i$ für $i = \max(i_1, i_2)$
  - $w = w_1 w_2 \in L_i L_i \subseteq L_{i+1} \subseteq L$

Beweis des Lemmas — Teil 2

wir wissen

- für jedes $i \in \mathbb{N}_0$: $\varepsilon \in L_i$
- für jedes $i \in \mathbb{N}_0$: $L_i \subseteq L_{i+1}$
- $L \subseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$

noch zu zeigen: $L \supseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$

- sei $w \in \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$.
- 1. Fall: $w = \varepsilon \in L_0 \subseteq L$.
- 2. Fall: $w \in LL$:
  - $w = w_1 w_2$ mit $w_1 \in L$ und $w_2 \in L$
  - $w_1 \in L_{i_1}$ und $w_2 \in L_{i_2}$
  - $w_1 \in L_i$ und $w_2 \in L_i$ für $i = \max(i_1, i_2)$
  - $w = w_1 w_2 \in L_i L_i \subseteq L_{i+1} \subseteq L$
- 3. Fall: $w \in \{(\}L\{)\}$:
  dann $w \in \{(\}L_i\{)\} \subseteq L_{i+1} \subseteq L$ für ein $i \in \mathbb{N}_0$

Lösung von $L = \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$ nicht eindeutig

$\{(,)\}^*$ ist auch eine Lösung, denn

- „$\subseteq$" zeigt man wie oben
- „$\supseteq$" ist trivial, da $\{(,)\}^*$ eben *alle* Wörter sind.

$\{(,)\}^*$ ist eine *andere* Lösung, denn

- $(((\,$ ist zwar in $\{(,)\}^*$
- aber *nicht* in $\bigcup_{i \in \mathbb{N}_0} L_i$:
  man vergleiche die Anzahlen der $($ und $)$

Was kann man an $\bigcup_{i\in\mathbb{N}_0} L_i$ sehen?

$$L_0 = \{\varepsilon\}$$

$$L_1 \setminus L_0 = \{\ ()\ \}$$

$$L_2 \setminus L_1 = \{\ ()()\ ,\ (())\ \}$$

$$L_3 \setminus L_2 = \{\ ()()()\ ,\ (())()\ ,\ ()(())\ ,$$

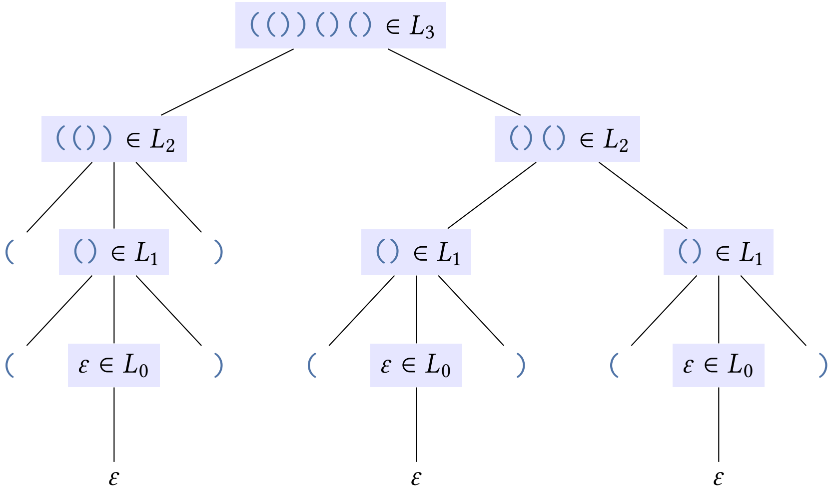
$$()()()()\ ,\ ()()(())\ ,\ (())()()\ ,\ (())(())\ ,$$

$$(()())\ ,\ ((()))\ \}$$

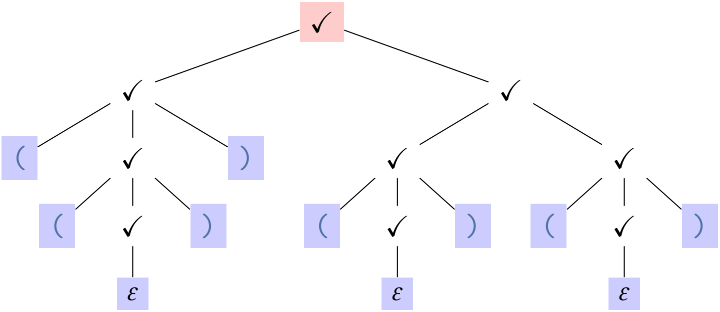
Dabei gilt z. B.:

- $(())()() \in L_3$, weil $(()) \in L_2$ und $()() \in L_2$ und Regel K2
  - $(()) \in L_2$, weil $() \in L_1$ und Regel K3.
    - $() \in L_1$, weil $\varepsilon \in L_0$ und Regel K3.
  - $()() \in L_2$, weil $() \in L_1$ und $() \in L_1$ ist und Regel K2
    - $() \in L_1$, weil $\varepsilon \in L_0$ und Regel K3.
    - $() \in L_1$, weil $\varepsilon \in L_0$ und Regel K3.

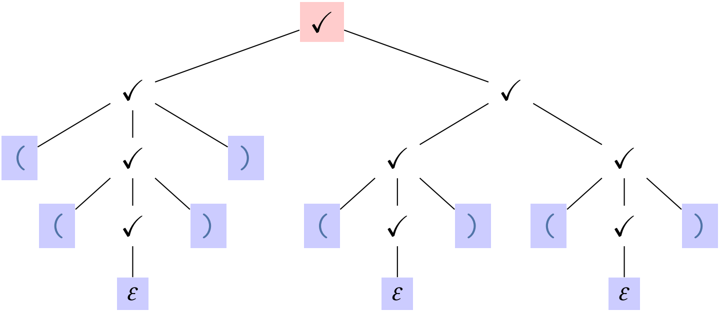
Die Erklärung für $(())()() \in L_3$ graphisch dargestellt



Vereinfachte Darstellung für $(())()() \in L_3$



Vereinfachte Darstellung für $(())()() \in L_3$



mal wieder ein Baum (Wurzel oben, Blätter unten)

bisher: von unten nach oben interpretiert als Begründungen

kontextfreie Grammatiken: von oben nach unten
syntaktische Ersetzungen

Was ist wichtig

Das sollten Sie mitnehmen:

- Klammerstrukturen sind wichtig.
- Manchmal kann man sich einem Fixpunkt «annähern».
  - Fixpunkt von f ist ein x mit $x = f(x)$.
  - *so* kann man $L = \{\varepsilon\} \cup LL \cup (L)$ auch sehen …

Das sollten Sie üben:

- keine Angst vor dem Lesen und Finden von Beweisen
  - ruhiges Hinsehen: eine passende Vorgehensweise drängt sich manchmal fast auf

Wo sind wir?

Rekursive Definition syntaktischer Strukturen

Kontextfreie Grammatiken

Relationen (Teil 2)

Eine Grenze kontextfreier Grammatiken

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

N Alphabet der *Nichtterminalsymbole*

T Alphabet der *Terminalsymbole*

- kein Zeichen in beiden Alphabeten: $N \cap T = \{\}$.

$S \in N$ das *Startsymbol*

$P \subseteq N \times V^*$ ist *endliche* Menge von *Produktionen*

- $V = N \cup T$ Menge aller Symbole überhaupt
- für jedes $(X, w) \in P$ schreibt man $X \to w$
- Bedeutung: man kann X ersetzen durch w

Beispiel

- $G = (\ \{X\},\ \{\mathtt{a}, \mathtt{b}\},\ X,\ \{X \to \varepsilon, X \to \mathtt{a}X\mathtt{b}\}\ )$

Ableitungsschritt mittels einer Produktion — Ersetzung der linken Seite durch die rechte

$u \Rightarrow v$

- aus $u \in V^*$ ist $v \in V^*$ in einem Schritt ableitbar
- wenn $u = w_1 X w_2$ und $v = w_1 w w_2$
- für Produktion $X \to w$ in P und Wörter $w_1, w_2 \in V^*$

Beispiel $G = (\{X\}, \{\mathtt{a}, \mathtt{b}\}, X, P)$ mit $P = \{X \to \varepsilon, X \to \mathtt{a}X\mathtt{b}\}$

Dann gilt z. B. $\mathtt{aba}X\mathtt{ba}XXXX \Rightarrow \mathtt{aba}X\mathtt{baa}X\mathtt{b}XXX$, denn

$$\underbrace{\mathtt{aba}X\mathtt{ba}}_{w_1}\, X\, \underbrace{XXX}_{w_2} \Rightarrow \underbrace{\mathtt{aba}X\mathtt{ba}}_{w_1}\, \mathtt{a}X\mathtt{b}\, \underbrace{XXX}_{w_2}$$

Ebenso gilt $\mathtt{aba}X\mathtt{ba}XXXX \Rightarrow \mathtt{abaa}X\mathtt{bba}XXXX$, denn

$$\underbrace{\mathtt{aba}}_{w_1}\, X\, \underbrace{\mathtt{ba}XXXX}_{w_2} \Rightarrow \underbrace{\mathtt{aba}}_{w_1}\, \mathtt{a}X\mathtt{b}\, \underbrace{\mathtt{ba}XXXX}_{w_2}$$

Anmerkungen (1)

bei einer Produktion

- linke Seite immer ein Nichtterminalsymbol
- Terminalsymbole werden nie ersetzt

Ersetzung $X \to w$ immer überall möglich

- unabhängig vom Kontext
- kontextfrei

Anmerkungen (2)

$\Rightarrow$ legt Relation zwischen Wörtern fest

- könnte auch schreiben: $R_{\Rightarrow} \subseteq V^* \times V^*$

Infixschreibweise üblich

- schreibe $u \Rightarrow v$ und nicht $(u, v) \in R_{\Rightarrow}$,
- analog zu $5 \leq 7$ und nicht $(5, 7) \in R_{\leq}$

Ableitungsrelation $\Rightarrow$ im allgemeinen

- weder links- noch rechtstotal
- weder links- noch rechtseindeutig

Vorsicht: $\Rightarrow$ nicht mit anderen Pfeilen verwechseln

Ableitungsfolgen werden $\Rightarrow^i$ und $\Rightarrow^*$ notiert

für jede $u, v \in V^*$ und für jedes $i \in \mathbb{N}_0$ gelte

$$u \Rightarrow^0 v \text{ genau dann, wenn } u = v$$
$$u \Rightarrow^{i+1} v \text{ genau dann, wenn für ein } w \in V^* : u \Rightarrow w \Rightarrow^i v$$
$$u \Rightarrow^* v \text{ genau dann, wenn für ein } i \in \mathbb{N}_0 : u \Rightarrow^i v$$

Beispiel $G = (\{X\}, \{\mathtt{a}, \mathtt{b}\}, X, \{X \to \varepsilon, X \to \mathtt{a}X\mathtt{b}\})$:

$$X \Rightarrow \mathtt{a}X\mathtt{b} \Rightarrow \mathtt{aa}X\mathtt{bb} \Rightarrow \mathtt{aaa}X\mathtt{bbb} \Rightarrow \mathtt{aaabbb}$$

- $X \Rightarrow^* \mathtt{aa}X\mathtt{bb}$
- $\mathtt{a}X\mathtt{b} \Rightarrow^* \mathtt{aaa}X\mathtt{bbb}$
- $X \Rightarrow^* \mathtt{aaabbb}$
- $\mathtt{abb} \Rightarrow^* \mathtt{abb}$

Jede Grammatik erzeugt eine formale Sprache

Welche Wörter aus T^* können
aus Startsymbol abgeleitet werden?

von $G = (N, T, S, P)$ erzeugte formale Sprache

$$L(G) = \{w \in T^* \mid S \Rightarrow^* w\}$$

solche formalen Sprachen heißen *kontextfrei*

Beispiel einer kontextfreien Grammatik/Sprache

Beispielgrammatik $G = (\{X\}, \{\mathtt{a}, \mathtt{b}\}, X, \{X \to \varepsilon, X \to \mathtt{a}X\mathtt{b}\}$

$\mathtt{aaabbb} \in L(G)$ wegen

- $X \Rightarrow \mathtt{a}X\mathtt{b} \Rightarrow \mathtt{aa}X\mathtt{bb} \Rightarrow \mathtt{aaa}X\mathtt{bbb} \Rightarrow \mathtt{aaabbb}$

für alle $i \in \mathbb{N}_0$ gilt: $X \Rightarrow \mathtt{a}^i\mathtt{b}^i$

- $\{\mathtt{a}^i\mathtt{b}^i \mid i \in \mathbb{N}_0\} \subseteq L(G)$

Beweis leichter, wenn man gleich zeigt:

- für jedes $i \in \mathbb{N}_0 : (X \Rightarrow^* \mathtt{a}^i\mathtt{b}^i \wedge X \Rightarrow^* \mathtt{a}^iX\mathtt{b}^i)$

Umgekehrt kann man zeigen:

- für jedes $i \in \mathbb{N}_0$: wenn $X \Rightarrow^{i+1} w$
  dann $w = \mathtt{a}^i\mathtt{b}^i$ oder $w = \mathtt{a}^{i+1}X\mathtt{b}^{i+1}$
- also $L(G) \subseteq \{\mathtt{a}^i\mathtt{b}^i \mid i \in \mathbb{N}_0\}$

Insgesamt:

$$L(G) = \{\mathtt{a}^i\mathtt{b}^i \mid i \in \mathbb{N}_0\} \ .$$

Kompaktere Notation bei vielen Produktionen

statt $\{X \to w_1, X \to w_2, X \to w_3, X \to w_4, X \to w_5\}$
schreibe man $\{X \to w_1|w_2|w_3|w_4|w_5\}$

und lese die senkrechten Striche als «oder»

Beispielgrammatik:

$$P = \{\ X \to \mathtt{a}X\mathtt{b} \mid \varepsilon\ \}$$

Java-Syntax — Interpretation der Definition

kontextfreie Grammatik

- $\langle Block \rangle, \ldots$ jeweils *ein* Nichtterminalsymbol
- Doppelpunkt entspricht Pfeil $\rightarrow$
- eingerückte Zeile: rechte Seite einer Produktion
- aufeinander folgende Zeilen denke man sich durch senkrechte Striche | getrennt
- Beispiel

```
2      BlockStatements:
               BlockStatement
               BlockStatements BlockStatement
```

- bedeutet

$$\begin{aligned} \langle BlockStatements \rangle \rightarrow & \langle BlockStatement \rangle \\ \mid\ & \langle BlockStatements \rangle \langle BlockStatement \rangle \end{aligned}$$

Java-Syntax — Interpretation der Definition (2)

«optionaler» Bestandteil

1	Block:	
		{ BlockStatements$_{opt}$ }

bedeutet

$$\langle \mathit{Block} \rangle \to \{ \; \langle \mathit{BlockStatements} \rangle \; \} \mid \{ \; \}$$

kontextfreie Grammatiken versus Syntax von Programmiersprachen

viele Nichtterminalsymbole stehen für strukturelle Konzepte der Programmiersprache

Ideal

- alles, was nicht ableitbar ist, ist syntaktisch falsch
- alles, was ableitbar ist, ist syntaktisch korrekt

Realität

- alles, was nicht ableitbar ist, ist syntaktisch falsch
- aber auch Dinge ableitbar, die syntaktisch falsch
  - manche Forderungen an syntaktische Korrekheit in «üblichen» Programmiersprachen nicht mit kontextfreien Grammatiken auszudrücken
  - aber «das meiste»

Ableitungsbäume sind übersichtlicher als schrittweise Ableitungen

Grammatik für unser Klammerproblem:
$(\{X\}, \{(,)\}, X, \{X \to XX | (X) | \varepsilon\})$.

lange Ableitungsfolgen manchmal nicht sehr erhellend:

$$X \Rightarrow XX \Rightarrow (X)X \Rightarrow (X)XX \Rightarrow (X)X(X) \Rightarrow ((X))X(X)$$
$$\Rightarrow ((X))X() \Rightarrow ((X))(X)() \Rightarrow (())(X)() \Rightarrow (())()($$

man darf umordnen (Kontextfreiheit!)

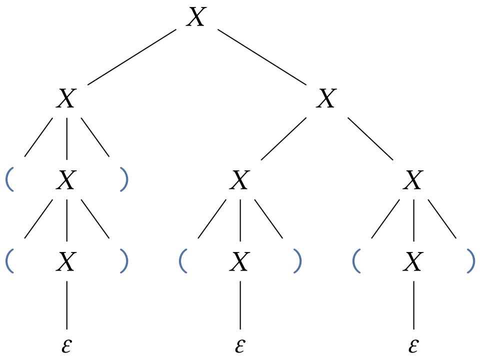
Linksableitung besser

$$X \Rightarrow XX \Rightarrow (X)X \Rightarrow ((X))X \Rightarrow (())X \Rightarrow (())XX$$
$$\Rightarrow (())(X)X \Rightarrow (())()X \Rightarrow (())()(X) \Rightarrow (())()()$$

Rechtsableitung analog

noch übersichtlicher: *Ableitungsbaum*

Ableitungsbaum



beginne mit Startsymbol als Wurzel

Ableitungsschritt $X \to w$

- von ersetztem X
- Kanten nach unten
- zu den Symbolen von w

Wohlgeformte/korrekte Klammerausdrücke

$(\{X\}, \{(,)\}, X, \{X \to XX|(X)|\varepsilon\})$ erzeugt sogenannte *wohlgeformte* oder *korrekte Klammerausdrücke* (KKA).

umgangssprachlich

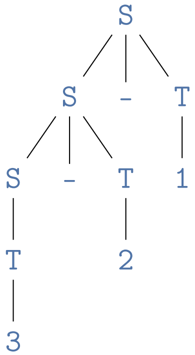
- was sich ergibt, wenn man in «normalen» arithmetischen Ausdrücken alles außer den Klammern weglässt

(etwas) präziser

- ε ist KKA.
- Wenn w_1, w_2 KKA sind, dann auch $w_1 w_2$.
- Wenn w KKA ist, dann auch (w).
- Nichts anderes ist KKA.

Klammersprachen sind ganz wesentliche Beispiele kontextfreier Sprachen

Arithmetische Ausdrücke



$G = (N, T, \mathtt{S}, P)$ die Grammatik mit

- Nichtterminalsymbolen $N = \{\mathtt{S}, \mathtt{T}\}$
- Terminalsymbolen $T = \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{/}, \mathtt{(}, \mathtt{)}\}$
- Startsymbol $\mathtt{S}$
- Produktionen $P = \{\mathtt{S} \to \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
  $\mathtt{T} \to \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3}\}.$

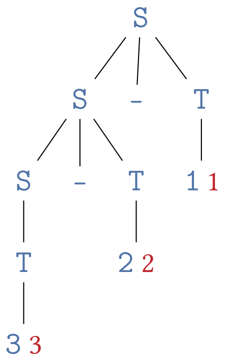
betrachte $\mathtt{3-2-1}$

- Ableitungsbaum links
- vergleiche $\mathtt{(3-2)-1}$ und $\mathtt{3-(2-1)}$
- ähnliches Problem: **if** (B_1) **if** (B_2) S_1 **else** S_2

Vorlesung Compilerbau

- Auswertung von Ausdrücken
- Erzeugung eines Programm(teil)s zur Auswertung von Ausdrücken

Arithmetische Ausdrücke



$G = (N, T, \texttt{S}, P)$ die Grammatik mit

- Nichtterminalsymbolen $N = \{\texttt{S}, \texttt{T}\}$
- Terminalsymbolen $T = \{\texttt{1}, \texttt{2}, \texttt{3}, \texttt{+}, \texttt{-}, \texttt{*}, \texttt{/}, \texttt{(}, \texttt{)}\}$
- Startsymbol $\texttt{S}$
- Produktionen $P = \{\texttt{S} \to \texttt{T} \mid \texttt{S+T} \mid \texttt{S-T} \mid \texttt{S*T} \mid \texttt{S/T},$
  $\texttt{T} \to \texttt{(S)} \mid \texttt{1} \mid \texttt{2} \mid \texttt{3}\}.$

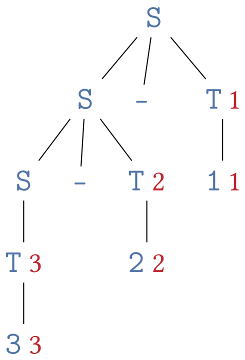
betrachte $\texttt{3-2-1}$

- Ableitungsbaum links
- vergleiche $\texttt{(3-2)-1}$ und $\texttt{3-(2-1)}$
- ähnliches Problem: **if** (B_1) **if** (B_2) S_1 **else** S_2

Vorlesung Compilerbau

- Auswertung von Ausdrücken
- Erzeugung eines Programm(teil)s zur Auswertung von Ausdrücken

Arithmetische Ausdrücke



$G = (N, T, \mathtt{S}, P)$ die Grammatik mit

- Nichtterminalsymbolen $N = \{\mathtt{S}, \mathtt{T}\}$
- Terminalsymbolen $T = \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{/}, \mathtt{(}, \mathtt{)}\}$
- Startsymbol $\mathtt{S}$
- Produktionen $P = \{\mathtt{S} \to \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
  $\mathtt{T} \to \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3}\}.$

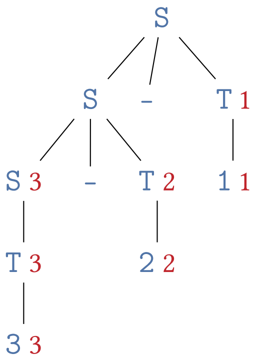
betrachte `3-2-1`

- Ableitungsbaum links
- vergleiche `(3-2)-1` und `3-(2-1)`
- ähnliches Problem: **if** (B_1) **if** (B_2) S_1 **else** S_2

Vorlesung Compilerbau

- Auswertung von Ausdrücken
- Erzeugung eines Programm(teil)s zur Auswertung von Ausdrücken

Arithmetische Ausdrücke



$G = (N, T, \mathtt{S}, P)$ die Grammatik mit

- Nichtterminalsymbolen $N = \{\mathtt{S}, \mathtt{T}\}$
- Terminalsymbolen $T = \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{/}, \mathtt{(}, \mathtt{)}\}$
- Startsymbol $\mathtt{S}$
- Produktionen $P = \{\mathtt{S} \to \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
  $\mathtt{T} \to \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3}\}.$

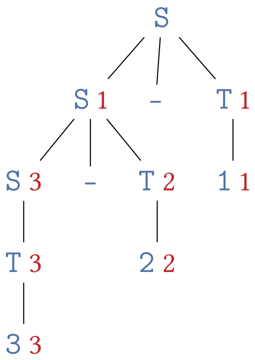
betrachte $\mathtt{3-2-1}$

- Ableitungsbaum links
- vergleiche $\mathtt{(3-2)-1}$ und $\mathtt{3-(2-1)}$
- ähnliches Problem: **if** (B_1) **if** (B_2) S_1 **else** S_2

Vorlesung Compilerbau

- Auswertung von Ausdrücken
- Erzeugung eines Programm(teil)s zur Auswertung von Ausdrücken

Arithmetische Ausdrücke



$G = (N, T, \mathtt{S}, P)$ die Grammatik mit

- Nichtterminalsymbolen $N = \{\mathtt{S}, \mathtt{T}\}$
- Terminalsymbolen $T = \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{/}, \mathtt{(}, \mathtt{)}\}$
- Startsymbol $\mathtt{S}$
- Produktionen $P = \{\mathtt{S} \to \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
  $\mathtt{T} \to \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3}\}.$

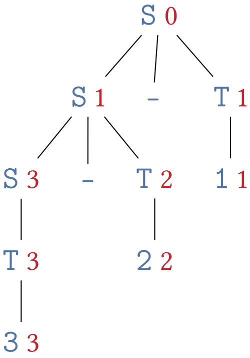
betrachte $\mathtt{3-2-1}$

- Ableitungsbaum links
- vergleiche $\mathtt{(3-2)-1}$ und $\mathtt{3-(2-1)}$
- ähnliches Problem: **if** (B_1) **if** (B_2) S_1 **else** S_2

Vorlesung Compilerbau

- Auswertung von Ausdrücken
- Erzeugung eines Programm(teil)s zur Auswertung von Ausdrücken

Arithmetische Ausdrücke



$G = (N, T, \mathtt{S}, P)$ die Grammatik mit

- Nichtterminalsymbolen $N = \{\mathtt{S}, \mathtt{T}\}$
- Terminalsymbolen $T = \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{/}, \mathtt{(}, \mathtt{)}\}$
- Startsymbol $\mathtt{S}$
- Produktionen $P = \{\mathtt{S} \to \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
  $\mathtt{T} \to \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3}\}.$

betrachte `3-2-1`

- Ableitungsbaum links
- vergleiche `(3-2)-1` und `3-(2-1)`
- ähnliches Problem: **if** (B_1) **if** (B_2) S_1 **else** S_2

Vorlesung Compilerbau

- Auswertung von Ausdrücken
- Erzeugung eines Programm(teil)s zur Auswertung von Ausdrücken

Syntax aussagenlogischer Formeln

leicht durch kontextfreie Grammatiken beschreibbar

$G = (\{X\}, T, X, P)$ mit

$T = Var_{AL} \cup \{(,), \neg, \wedge, \vee, \rightarrow\}$

- $Var_{AL} \subseteq \{\mathtt{P}_i \mid i \in \mathbb{N}_0\}$, endlich

Produktionen auf der nächsten Folie

Achtung: Verwechslungsgefahr bei Pfeilen

- der Pfeil $\rightarrow$ in aussagenlogischen Formeln
- der Pfeil $\rightarrow$ zwischen linker und rechter Seite von Produktionen

und bei senkrechten Strichen

- in set comprehensions
- als Trenner zwischen rechten Seiten von Produktionen

Syntax aussagenlogischer Formeln — die Produktionen

$G = (\{X\}, T, X, P)$ mit

- $T = Var_{AL} \cup \{(,), \neg, \wedge, \vee, \rightarrow\}$
- $Var_{AL} \subseteq \{\mathtt{P}_i \mid i \in \mathbb{N}_0\}$, endlich

Produktionenmenge

$$P = \{ X \rightarrow \mathtt{P}_i \mid \mathtt{P}_i \in Var_{AL}\} \cup \left\{ \begin{array}{ll} X \rightarrow & (\neg X) \\ & \mid (X \wedge X) \\ & \mid (X \vee X) \\ & \mid (X \rightarrow X) \end{array} \right\}$$

Aus X sind genau die aussagenlogischen Formeln mit Aussagevariablen in Var_{AL} ableitbar.

Was ist wichtig

Das sollten Sie mitnehmen:

- kontextfreie Grammatik
- Ableitung
- erzeugte formale Sprache
- Ableitungsbaum

Das sollten Sie üben:

- (semi-)reale Produktionenmengen lesen (Java, ...)
- zu formaler Sprache sie
  erzeugende kontextfreie Grammatik konstruieren
- zu kontextfreier Grammatik die
  erzeugte formale Sprache bestimmen

Wo sind wir?

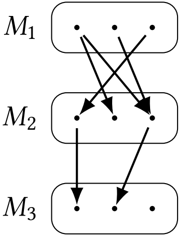
Rekursive Definition syntaktischer Strukturen

Kontextfreie Grammatiken

Relationen (Teil 2)

Eine Grenze kontextfreier Grammatiken

Produkt von Relationen



Es seien $R \subseteq M_1 \times M_2$ und $S \subseteq M_2 \times M_3$ zwei Relationen

Produkt der Relationen R und S

- $S \circ R = \{(x, z) \in M_1 \times M_3 \mid$ es gibt $y \in M_2 : (x, y) \in R \wedge (y, z) \in S\}$

Infixschreibweise: für alle $(x, z) \in M_1 \times M_3$

- $x(S \circ R)z \quad$ gdw. $\quad \exists y \in M_2 : xRy \wedge ySz$

$\circ$ ist assoziativ

- $(R \circ S) \circ T = R \circ (S \circ T)$

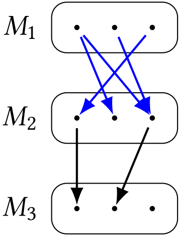
die identische Abbildung auf M als Relation I_M

- $\mathrm{I}_M = \{(x, x) \mid x \in M\}$

I_M neutrale Elemente bzgl. $\circ$

- $R \circ \mathrm{I}_{M_1} = R = \mathrm{I}_{M_2} \circ R$

Produkt von Relationen



Es seien $R \subseteq M_1 \times M_2$ und $S \subseteq M_2 \times M_3$ zwei Relationen

Produkt der Relationen R und S

- $S \circ R = \{(x, z) \in M_1 \times M_3 \mid$ es gibt $y \in M_2 : (x, y) \in R \wedge (y, z) \in S\}$

Infixschreibweise: für alle $(x, z) \in M_1 \times M_3$

- $x(S \circ R)z \quad$ gdw. $\quad \exists y \in M_2 : xRy \wedge ySz$

$\circ$ ist assoziativ

- $(R \circ S) \circ T = R \circ (S \circ T)$

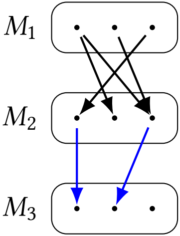
die identische Abbildung auf M als Relation I_M

- $\mathrm{I}_M = \{(x, x) \mid x \in M\}$

I_M neutrale Elemente bzgl. $\circ$

- $R \circ \mathrm{I}_{M_1} = R = \mathrm{I}_{M_2} \circ R$

Produkt von Relationen



Es seien $R \subseteq M_1 \times M_2$ und $S \subseteq M_2 \times M_3$ zwei Relationen

Produkt der Relationen R und S

- $S \circ R = \{(x,z) \in M_1 \times M_3 \mid$ es gibt $y \in M_2 : (x,y) \in R \wedge (y,z) \in S\}$

Infixschreibweise: für alle $(x,z) \in M_1 \times M_3$

- $x(S \circ R)z$ gdw. $\exists y \in M_2 : xRy \wedge ySz$

$\circ$ ist assoziativ

- $(R \circ S) \circ T = R \circ (S \circ T)$

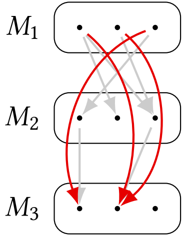
die identische Abbildung auf M als Relation I_M

- $\mathrm{I}_M = \{(x,x) \mid x \in M\}$

I_M neutrale Elemente bzgl. $\circ$

- $R \circ \mathrm{I}_{M_1} = R = \mathrm{I}_{M_2} \circ R$

Produkt von Relationen



Es seien $R \subseteq M_1 \times M_2$ und $S \subseteq M_2 \times M_3$ zwei Relationen

Produkt der Relationen R und S

- $S \circ R = \{(x, z) \in M_1 \times M_3 \mid$ es gibt $y \in M_2 : (x, y) \in R \land (y, z) \in S\}$

Infixschreibweise: für alle $(x, z) \in M_1 \times M_3$

- $x(S \circ R)z$ gdw. $\exists y \in M_2 : xRy \land ySz$

$\circ$ ist assoziativ

- $(R \circ S) \circ T = R \circ (S \circ T)$

die identische Abbildung auf M als Relation I_M

- $\mathrm{I}_M = \{(x, x) \mid x \in M\}$

I_M neutrale Elemente bzgl. $\circ$

- $R \circ \mathrm{I}_{M_1} = R = \mathrm{I}_{M_2} \circ R$

Reflexiv-transitive Hülle einer Relation — Vereinigung aller Potenzen

Ist $R \subseteq M \times M$ binäre Relation auf M, dann definiert man *Potenzen* R^i:

$$R^0 = \mathrm{I}_M$$
$$\text{für jedes } i \in \mathbb{N}_0 : R^{i+1} = R^i \circ R$$

Die *reflexiv-transitive Hülle* einer Relation R ist

$$R^* = \bigcup_{i \in \mathbb{N}_0} R^i$$

Reflexiv-transitive Hülle einer Relation — ein Beispiel

es sei $R = \{(n, n+1) \mid n \in \mathbb{N}_0\} \subseteq \mathbb{N}_0 \times \mathbb{N}_0$

dann

$$\begin{aligned} R \circ R &= \{(n, m) \mid \text{es gibt } k \in \mathbb{N}_0 \colon (n, k) \in R \text{ und } (k, m) \in R\} \\ &= \{(n, m) \mid \text{es gibt } k \in \mathbb{N}_0 \colon k = n + 1 \text{ und } m = k + 1 \in R\} \\ &= \{(n, n+2) \mid \text{es gibt } n \in \mathbb{N}_0\} \end{aligned}$$

also

$$\begin{aligned} R^0 &= \{(n, n) \mid n \in \mathbb{N}_0\} \\ R^1 &= \{(n, n+1) \mid n \in \mathbb{N}_0\} \\ R^2 &= \{(n, n+2) \mid n \in \mathbb{N}_0\} \end{aligned}$$

usw.

also $R^* = \{(n, m) \mid n \leq m\}$

Eigenschaften der reflexiv-transitiven Hülle

überlegen uns gleich:

- R^* ist reflexiv.
- R^* ist transitiv.
- R^* ist kleinste Relation, die
  R enthält und reflexiv und transitiv ist.

Relation R heißt *reflexiv*, wenn $\mathrm{I}_M \subseteq R$, also wenn

- für jedes $x \in M$: $x \; R \; x$

Relation R heißt *transitiv*, wenn $R \circ R \subseteq R$, also wenn:

- für jedes $x \in M$, jedes $y \in M$, jedes $z \in M$:
  wenn $x \; R \; y$ und $y \; R \; z$, dann $x \; R \; z$

Eigenschaften der reflexiv-transitiven Hülle — Erläuterungen

R^* ist immer reflexiv

- $\mathrm{I}_M = R^0 \subseteq R^*$

für jedes $i, j \in \mathbb{N}_0$ gilt: $R^i \circ R^j = R^{i+j}$.

R^* ist immer transitiv, denn

- wenn $(x, y) \in R^*$ und $(y, z) \in R^*$,
- dann $(x, y) \in R^i$ und $(y, z) \in R^j$ für i und $j \in \mathbb{N}_0$
- dann $(x, z) \in R^i \circ R^j = R^{i+j} \subseteq R^*$.

R^* ist die *kleinste* Relation, die R umfasst und reflexiv und transitiv ist:

- R^* umfasst R und ist reflexiv und transitiv.
- wenn S beliebige Relation, die reflexiv und transitiv ist, und
- wenn $R \subseteq S$, dann sogar $R^* \subseteq S$.

Was ist wichtig

Das sollten Sie mitnehmen:

- Produkte und Potenzen von Relationen
- reflexive und transitive Relationen
- reflexiv-transitive Hülle einer Relation
  - «klassisches» Beispiel: Ableitbarkeit $\Rightarrow^*$

Das sollten Sie üben:

- Transitivität nachweisen
- Bilder von Relationen malen

Wo sind wir?

Rekursive Definition syntaktischer Strukturen

Kontextfreie Grammatiken

Relationen (Teil 2)

Eine Grenze kontextfreier Grammatiken

Eine Grenze kontextfreier Grammatiken

viele Sprachen in der Informatik sind kontextfrei

- jedenfalls „ein Großteil“
- und der sogar mit zusätzlichen Einschränkungen
- Details in Vorlesungen zu Übersetzerbau

aber es gibt auch „kompliziertere“ Sprachen

Eine Grenze kontextfreier Grammatiken

viele Sprachen in der Informatik sind kontextfrei

- jedenfalls „ein Großteil“
- und der sogar mit zusätzlichen Einschränkungen
- Details in Vorlesungen zu Übersetzerbau

aber es gibt auch „kompliziertere“ Sprachen

Lemma. Es gibt keine kontextfreie Grammatik, die die formale Sprache über $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

$$L_{vv} = \{v\mathtt{c}v \mid v \in \{\mathtt{a}, \mathtt{b}\}^*\}$$

erzeugt.

Beweisskizze auf den folgenden Folien

L_{vv} — Beispielwörter

$L_{vv} = \{v\mathtt{c}v \mid v \in \{\mathtt{a}, \mathtt{b}\}^*\}$

zu L_{vv} gehören zum Beispiel

- aabacaaba
- aca
- bbbcbbb

zu L_{vv} gehören zum Beispiel *nicht*

- aabacaabb
- aaacaaaaaa
- abaccaba
- aaacbcaaa
- aabaab

L_{vv} ist nicht kontextfrei

Hilfsmittel (ohne Beweis):

Lemma. Es sei $G = (N, T, S, P)$ eine kontextfreie Grammatik mit $\varepsilon \notin L(G)$.

Dann gibt es eine kontextfreie Grammatik $G' = (N', T, S', P')$ mit

- $L(G') = L(G)$ und
- in P' hat keine Produktion ε als rechte Seite.

nennen wir eine solche Grammatik *ε-frei*

L_{vv} ist nicht kontextfrei

Hilfsmittel (ohne Beweis):

Lemma. Es sei $G = (N, T, S, P)$ eine kontextfreie Grammatik mit $\varepsilon \notin L(G)$.

Dann gibt es eine kontextfreie Grammatik $G' = (N', T, S', P')$ mit

- $L(G') = L(G)$ und
- in P' hat keine Produktion ε als rechte Seite.

nennen wir eine solche Grammatik *ε-frei*

müssen daher „nur“ zeigen:

Keine ε-freie kf. Grammatik G erzeugt $L(G) = L_{vv}$.

L_{vv} ist nicht kontextfrei

Hilfsmittel (ohne Beweis):

Lemma. Es sei $G = (N, T, S, P)$ eine kontextfreie Grammatik mit $\varepsilon \notin L(G)$.

Dann gibt es eine kontextfreie Grammatik $G' = (N', T, S', P')$ mit

- $L(G') = L(G)$ und
- in P' hat keine Produktion ε als rechte Seite.

nennen wir eine solche Grammatik *ε-frei*

müssen daher „nur“ zeigen:

Keine ε-freie kf. Grammatik G erzeugt $L(G) = L_{vv}$. d. h.
Jede ε-freie kf. Grammatik G erzeugt $L(G) \neq L_{vv}$.

L_{vv} ist nicht kontextfrei

Hilfsmittel (ohne Beweis):

Lemma. Es sei $G = (N, T, S, P)$ eine kontextfreie Grammatik mit $\varepsilon \notin L(G)$.

Dann gibt es eine kontextfreie Grammatik $G' = (N', T, S', P')$ mit

- $L(G') = L(G)$ und
- in P' hat keine Produktion ε als rechte Seite.

nennen wir eine solche Grammatik *ε-frei*

Einschränkung der Menge zu betrachtender Objekte

müssen daher „nur" zeigen:

Keine ε-freie kf. Grammatik G erzeugt $L(G) = L_{vv}$. d. h.

Jede ε-freie kf. Grammatik G erzeugt $L(G) \neq L_{vv}$.

L_{vv} ist nicht kontextfrei (2)

Lemma.

Es sei

- $G = (N, T, S, P)$ eine ε-freie kf. Grammatik
- ℓ die größte Länge der rechten Seite einer Produktion
- $u, w, u' \in T^*$ mit $|w| > \ell^k$ für ein $k \in \mathbb{N}_0$

Wenn für ein $X \in N$

- $X \Longrightarrow^* uwu'$

dann

- liegen im Ableitungsbaum für $X \Longrightarrow^* uwu'$
- auf dem Weg von der Wurzel X zu mindestens einem Terminalsymbol von w
- mindestens $k + 1$ Nichtterminalsymbole
- *von denen jedes mindestens zwei Nachfolger hat.*

L_{vv} ist nicht kontextfrei (2) — Beweisskizze des Lemmas

Streiche im Ableitungsbaum alle Kanten und Knoten, die ausschließlich zu Symbolen aus u oder u' führen.

- (kein Ableitungsbaum mehr)

Die verbliebenen Blätter stehen für die Symbole von w.

Fasse alle längeren Pfade ohne Verzweigungen zu einer Kante zusammen.

Im resultierenden Baum hat jeder innere Knoten einen Verzweigungsgrad zwischen 2 und ℓ.

Wenn jeder Pfad von der Wurzel zu einem Blatt höchstens Länge k hätte,

dann könnte der Baum höchstens ℓ^k Blätter haben.

Aber $|w| > \ell^k$.

L_{vv} ist nicht kontextfrei (3)

zu zeigen: L_{vv} ist nicht kontextfrei

zeige:

- Es sei G eine ε-freie kf. Grammatik.
- Dann ist $L(G) \neq L_{vv}$.

L_{vv} ist nicht kontextfrei (3)

zu zeigen: L_{vv} ist nicht kontextfrei

zeige:

- Es sei G eine ε-freie kf. Grammatik.
- Dann ist $L(G) \neq L_{vv}$.

Was bedeutet $L(G) \neq L_{vv}$?

- *nicht* $(L_{vv} \subseteq L(G)$ und $L(G) \subseteq L_{vv})$

L_{vv} ist nicht kontextfrei (3)

zu zeigen: L_{vv} ist nicht kontextfrei

zeige:

- Es sei G eine ε-freie kf. Grammatik.
- Dann ist $L(G) \neq L_{vv}$.

Was bedeutet $L(G) \neq L_{vv}$?

- *nicht* $(L_{vv} \subseteq L(G)$ und $L(G) \subseteq L_{vv})$

also

- *nicht* $L_{vv} \subseteq L(G)$ oder
- *nicht* $L(G) \subseteq L_{vv}$

L_{vv} ist nicht kontextfrei (3)

zu zeigen: L_{vv} ist nicht kontextfrei

zeige:

- Es sei G eine ε-freie kf. Grammatik.
- Dann ist $L(G) \neq L_{vv}$.

Was bedeutet $L(G) \neq L_{vv}$?

- *nicht* $(L_{vv} \subseteq L(G)$ und $L(G) \subseteq L_{vv})$

also

- *nicht* $L_{vv} \subseteq L(G)$ oder
- *nicht* $L(G) \subseteq L_{vv}$

mache Fallunterscheidung

- 1. Fall: *nicht* $L_{vv} \subseteq L(G)$
- 2. Fall: $L_{vv} \subseteq L(G)$

L_{vv} ist nicht kontextfrei (3)

zu zeigen: L_{vv} ist nicht kontextfrei

zeige:

- Es sei G eine ε-freie kf. Grammatik.
- Dann ist $L(G) \neq L_{vv}$.

Was bedeutet $L(G) \neq L_{vv}$?

- *nicht* $(L_{vv} \subseteq L(G)$ und $L(G) \subseteq L_{vv})$

also

- *nicht* $L_{vv} \subseteq L(G)$ oder
- *nicht* $L(G) \subseteq L_{vv}$

mache Fallunterscheidung

- 1. Fall: *nicht* $L_{vv} \subseteq L(G)$
  - dann $L(G) \neq L_{vv}$ *fertig*
- 2. Fall: $L_{vv} \subseteq L(G)$

L_{vv} ist nicht kontextfrei (3)

zu zeigen: L_{vv} ist nicht kontextfrei

zeige:

- Es sei G eine ε-freie kf. Grammatik.
- Dann ist $L(G) \neq L_{vv}$.

Was bedeutet $L(G) \neq L_{vv}$?

- *nicht* $(L_{vv} \subseteq L(G)$ und $L(G) \subseteq L_{vv})$

also

- *nicht* $L_{vv} \subseteq L(G)$ oder
- *nicht* $L(G) \subseteq L_{vv}$

mache Fallunterscheidung

- 1. Fall: *nicht* $L_{vv} \subseteq L(G)$
  - dann $L(G) \neq L_{vv}$ *fertig*
- 2. Fall: $L_{vv} \subseteq L(G)$
  - zeige: dann *nicht* $L(G) \subseteq L_{vv}$

L_{vv} ist nicht kontextfrei (3)

zu zeigen: L_{vv} ist nicht kontextfrei

zeige:

- Es sei G eine ε-freie kf. Grammatik.
- Dann ist $L(G) \neq L_{vv}$.

Was bedeutet $L(G) \neq L_{vv}$?

- *nicht* $(L_{vv} \subseteq L(G)$ und $L(G) \subseteq L_{vv})$

also

- *nicht* $L_{vv} \subseteq L(G)$ oder
- *nicht* $L(G) \subseteq L_{vv}$

mache Fallunterscheidung

- 1. Fall: *nicht* $L_{vv} \subseteq L(G)$
  - dann $L(G) \neq L_{vv}$ *fertig*
- 2. Fall: $L_{vv} \subseteq L(G)$
  - zeige: dann *nicht* $L(G) \subseteq L_{vv}$

Einschränkung der Menge zu betrachtender Objekte

L_{vv} ist nicht kontextfrei (4)

verbliebene Aufgabe:

- Es sei $G = (N, \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}, S, P)$ eine ε-freie kf. Grammatik und
- $L_{vv} \subseteq L(G)$
- zeige: *nicht* $L(G) \subseteq L_{vv}$

L_{vv} ist nicht kontextfrei (4)

verbliebene Aufgabe:

- Es sei $G = (N, \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}, S, P)$ eine ε-freie kf. Grammatik und
- $L_{vv} \subseteq L(G)$
- zeige: *nicht* $L(G) \subseteq L_{vv}$ d. h.
  G erzeugt ein Wort $w' \notin L_{vv}$

L_{vv} ist nicht kontextfrei (4)

verbliebene Aufgabe:

- Es sei $G = (N, \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}, S, P)$ eine ε-freie kf. Grammatik und
- $L_{vv} \subseteq L(G)$
- zeige: *nicht* $L(G) \subseteq L_{vv}$ d. h.
  G erzeugt ein Wort $w' \notin L_{vv}$

es sei $k = |N|$

betrachte Ableitungsbaum für

$$w = \mathtt{a}^{1+\ell^{2k}} \mathtt{b}^{1+\ell^{2k}} \mathtt{c} \mathtt{a}^{1+\ell^{2k}} \mathtt{b}^{1+\ell^{2k}} \in L_{vv}$$

L_{vv} ist nicht kontextfrei (4)

verbliebene Aufgabe:

- Es sei $G = (N, \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}, S, P)$ eine ε-freie kf. Grammatik und
- $L_{vv} \subseteq L(G)$
- zeige: *nicht* $L(G) \subseteq L_{vv}$ d. h.
  G erzeugt ein Wort $w' \notin L_{vv}$

es sei $k = |N|$

betrachte Ableitungsbaum für

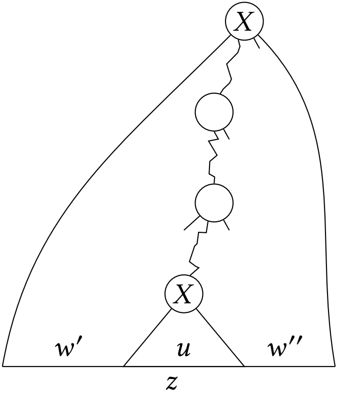
$$w = \mathtt{a}^{1+\ell^{2k}}\mathtt{b}^{1+\ell^{2k}}\mathtt{c}\mathtt{a}^{1+\ell^{2k}}\mathtt{b}^{1+\ell^{2k}} \in L_{vv}$$

nach bewiesenem Lemma

- auf Pfad zu mindestens einem Symbol mindestens
  $k + 1 = |N| + 1$ Nichtterminalsymbole mit Verzweigung
- also ein Nichtterminalsymbol doppelt auf Pfad

betrachte Terminal mit maximal vielen solchen N.sym.

L_{vv} ist nicht kontextfrei (5)



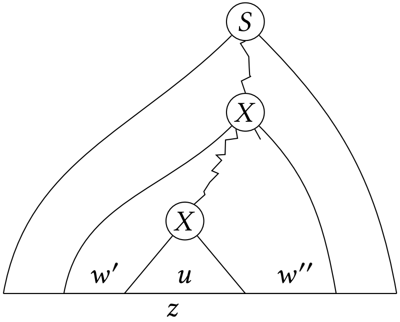
betrachte T.sym. z mit maximal vielen „verzweigenden“ N.sym.

- da G ε-frei erzeugt jede Verzweigung mind. ein Symbol

gehe von z aufwärts, bis sich erstmals ein „verzweigendes“ N.sym. X wiederholt

- aus unterstem Vorkommen Wort u abgeleitet mit $|u| < \ell^k$
- aus zweitunterstem Vorkommen Wort $w'uw''$ abgeleitet mit $|w'uw''| < \ell^{2k}$

L_{vv} ist nicht kontextfrei (6)



Wo steht das c?

- Außerhalb von $w'uw''$?
  - leite $\cdots w'w'uw''w'' \cdots$ ab
  - dann vor dem c
    mehr Symbole als dahinter
- In $w'w''$?
  - lasse den Teil der Ableitung weg
  - dann kein c mehr vorhanden
- In u?
  - leite $\cdots w'w'uw''w'' \cdots$ ab
  - dann vor c mehr b als dahinter

in allen Fällen Wort $\notin L_{vv}$ abgeleitet

Zusammenfassung

rekursive Definitionen syntaktischer Strukturen

- Vorsicht kann nicht schaden
- zumindest manchmal sinnvolle Interpretation möglich

Grammatiken

- kontextfreie Grammatik
- Ableitung
- erzeugte formale Sprache
- Ableitungsbaum

Relationen

- Produkte und Potenzen von Relationen
- reflexive und transitive Relationen
- reflexiv-transitive Hülle einer Relation