Grundbegriffe der Informatik

Kapitel 12: kontextfreie Grammatiken

Mattias Ulbrich
(basierend auf Folien von Thomas Worsch)

KIT · Institut für Theoretische Informatik

Wintersemester 2023/2024

In diesem Kapitel

Noch ein Beispiel für die
rekursive Definition syntaktischer Strukturen
soll andeuten, warum sich

kontextfreie Grammatiken
als sehr nützlich erwiesen haben.

Auf weitere wichtige Begriffe im Zusammenhang mit
Relationen kommen wir anschließend zu sprechen.

Wo sind wir?

Rekursive Definition syntaktischer Strukturen

Kontextfreie Grammatiken

Relationen (Teil 2)

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 (siehe Kapitel 18)

Ausschnitt der Definition der Syntax von Java

1	Block:	
		{ BlockStatements$_{opt}$ }
2	BlockStatements:	
		BlockStatement
		BlockStatements BlockStatement
3	BlockStatement:	
		Statement
		……
4	Statement:	
		StatementWithoutTrailingSubstatement
		……
5	StatementWithoutTrailingSubstatement:	
		Block
		……

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

Rekursion

- Definition von $\langle \mathit{BlockStatements} \rangle$ nimmt Bezug auf $\langle \mathit{BlockStatements} \rangle$
- Definition von $\langle \mathit{Block} \rangle$ nimmt (indirekt) Bezug auf $\langle \mathit{Statement} \rangle$
- Definition von $\langle \mathit{Statement} \rangle$ nimmt (indirekt) Bezug auf auf $\langle \mathit{Block} \rangle$
- 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: es ist 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 $L \supseteq \dots$ spiegelt K1, K2, K3 wider
  - die Inklusion $L \subseteq \dots$ 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 $L \supseteq \dots$ spiegelt K1, K2, K3 wider
  - die Inklusion $L \subseteq \dots$ 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
     $\rightsquigarrow$ finde und charakterisiere die «interessierende» Lösung

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

- konstruiere Folge $L_0, L_1, \dots$ 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 $\varepsilon = \varepsilon\varepsilon \in L_i L_i \subseteq L_{i+1}$.

„monotone Mengenfolge“

Beweis des Lemmas — Teil 1

„monotone Mengenfolge“

- 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 $\varepsilon = \varepsilon\varepsilon \in L_i L_i \subseteq L_{i+1}$.

„monotone Mengenfolge“

- 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

„monotone Mengenfolge“

- 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 $L = L\{\varepsilon\} \subseteq LL \subseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$.
- Zeige: $L \supseteq \{\varepsilon\} \cup LL \cup \{(\}L\{)\}$
  - 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

Karlsruher Institut für Technologie

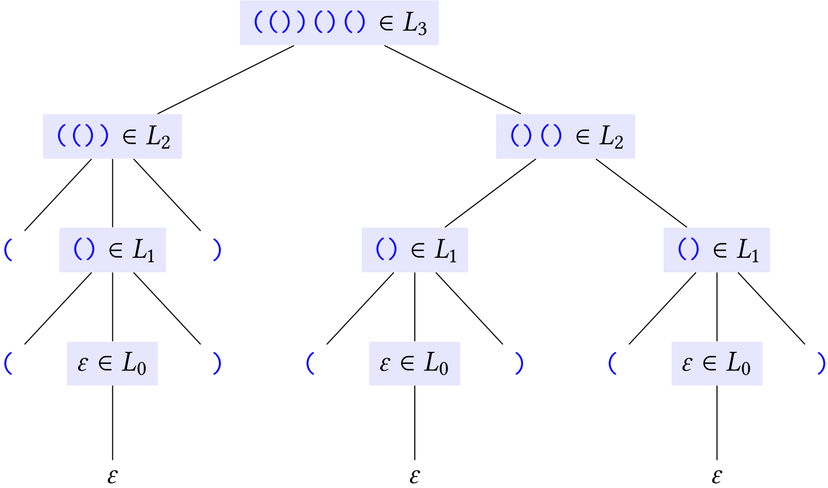
- $\{(,)\}^*$ ist auch eine Lösung, denn
  - „$\subseteq$" zeigt man wie oben
  - „$\supseteq$" ist trivial, da $\{(,)\}^*$ eben *alle* Wörter sind.
- $\{(,)\}^*$ ist eine *andere* Lösung, denn
  - $(((\,$ ist zwar in $\{(,)\}^*$
  - aber *nicht* in $\bigcup_{i \in \mathbb{N}_0} L_i$:
    man vergleiche die Anzahlen der $($ und $)$

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

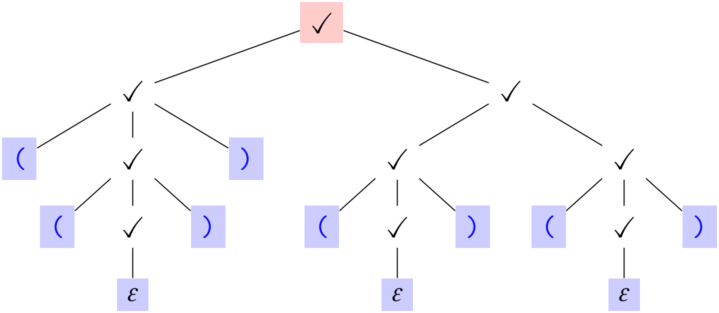
$$
\begin{aligned}
L_0 &= \{\varepsilon\} \\
L_1 \setminus L_0 &= \{ \; () \; \} \\
L_2 \setminus L_1 &= \{ \; ()() \; , \; (()) \; \} \\
L_3 \setminus L_2 &= \{ \; ()()() \; , \; (())() \; , \; ()(()) \; , \\
&\qquad ()()()() \; , \; ()()(()) \; , \; (())()() \; , \; (())(()) \; , \\
&\qquad (()()) \; , \; ((())) \; \}
\end{aligned}
$$

- 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.

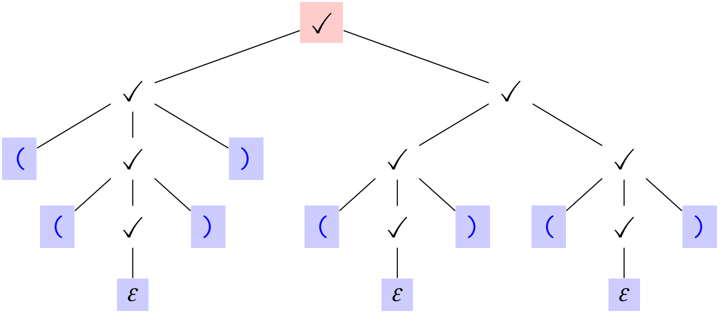
Die Erklärung für $(())()() \in L_3$ graphisch dargestellt



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

Karlsruher Institut für Technologie

- 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
    - in Ruhe hingucken:
      passende Vorgehensweise drängt sich manchmal fast auf

Wo sind wir?

Rekursive Definition syntaktischer Strukturen

Kontextfreie Grammatiken

Relationen (Teil 2)

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

Karlsruher Institut für Technologie

- 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 $w_1, w_2 \in V^*$
- $G = (\{X\}, \{\mathtt{a}, \mathtt{b}\}, X, P)$ 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 $X \longrightarrow w$ immer überall möglich
  - unabhängig vom Kontext
  - kontextfrei
- wenn $X \longrightarrow w \in P$
  dann $w_1 X w_2 \Longrightarrow w_1 w w_2$

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
  - nicht linkstotal, nicht rechtstotal
  - nicht linkseindeutig, nicht rechtseindeutig
- Vorsicht: $\Rightarrow$ nicht mit anderen Pfeilen verwechseln

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

Karlsruher Institut für Technologie

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

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

- Beispiel $G = (\{X\}, \{\mathtt{a}, \mathtt{b}\}, X, \{X \to \varepsilon, X \to \mathtt{a}X\mathtt{b}\})$:
  - $X \Rightarrow \mathtt{a}X\mathtt{b} \Rightarrow \mathtt{aa}X\mathtt{bb} \Rightarrow \mathtt{aaa}X\mathtt{bbb} \Rightarrow \mathtt{aaabbb}$
  - $X \Rightarrow^2 \mathtt{aa}X\mathtt{bb}$ also $X \Rightarrow^* \mathtt{aa}X\mathtt{bb}$
  - $\mathtt{a}X\mathtt{b} \Rightarrow^2 \mathtt{aaa}X\mathtt{bbb}$ also $\mathtt{a}X\mathtt{b} \Rightarrow^* \mathtt{aaa}X\mathtt{bbb}$
  - $X \Rightarrow^4 \mathtt{aaabbb}$ also $X \Rightarrow^* \mathtt{aaabbb}$
  - $\mathtt{abb} \Rightarrow^0 \mathtt{abb}$ also $\mathtt{abb} \Rightarrow^* \mathtt{abb}$

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: $X \Rightarrow^* \mathtt{a}^i\mathtt{b}^i$, also $\{\, \mathtt{a}^i\mathtt{b}^i \mid i \in \mathbb{N}_0 \,\} \subseteq L(G)$
  - Beweis leichter, wenn man gleich zeigt:
    für jedes $i \in \mathbb{N}_0 : (X \Rightarrow^* \mathtt{a}^i\mathtt{b}^i \land X \Rightarrow^* \mathtt{a}^iX\mathtt{b}^i)$

Beispiel einer kontextfreien Grammatik/Sprache

Karlsruher Institut für Technologie

- 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: $X \Rightarrow^* \mathtt{a}^i\mathtt{b}^i$, also $\{\, \mathtt{a}^i\mathtt{b}^i \mid i \in \mathbb{N}_0 \,\} \subseteq L(G)$
  - 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 $\{X \to w_1|w_2|w_3|w_4|w_5\}$
- 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 $\longrightarrow$ in Produktion
- eingerückte Zeile: rechte Seite einer Produktion
- aufeinander folgende Zeilen: verschiedene rechte Seiten
- Beispiel

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

- bedeutet

$$\langle BlockStatements \rangle \longrightarrow \langle BlockStatement \rangle \mid \langle BlockStatements \rangle \langle BlockStatement \rangle$$

Java-Syntax — Interpretation der Definition (2)

- «optionaler» Bestandteil

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

- bedeutet

$$\langle \mathit{Block} \rangle \rightarrow \texttt{\{}\ \langle \mathit{BlockStatements} \rangle\ \texttt{\}} \mid \texttt{\{}\ \texttt{\}}$$

Kontextfreie Grammatiken versus Syntax von Programmiersprachen

Karlsruher Institut für Technologie

- viele Nichtterminalsymbole stehen für strukturelle Konzepte der Programmiersprache
- Ideal
  - alles, was nicht ableitbar ist, ist syntaktisch falsch
  - alles, was ableitbar ist, ist syntaktisch korrekt

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 Klammersprache: $(\{X\}, \{(,)\}, X, \{X \rightarrow XX \mid (X) \mid \varepsilon\})$.
- lange Ableitungsfolgen manchmal nicht sehr erhellend:

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

Ableitungsbäume sind übersichtlicher als schrittweise Ableitungen

- Grammatik für Klammersprache: $(\{X\}, \{(,)\}, X, \{ X \to XX \mid (X) \mid \varepsilon \})$.
- lange Ableitungsfolgen manchmal nicht sehr erhellend:

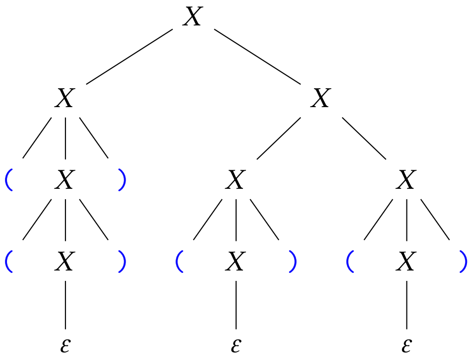
$$X \Rightarrow XX \Rightarrow (X)X \Rightarrow (X)XX \Rightarrow (X)X(X) \Rightarrow ((X))X(X)$$
$$\Rightarrow ((X))X() \Rightarrow ((X))(X)() \Rightarrow (())(X)() \Rightarrow (())()()$$

- man darf Reihenfolge der Ersetzungen 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

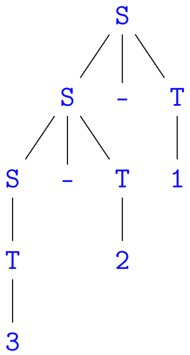


- beginne mit Startsymbol als Wurzel
- Ableitungsschritt $X \to w$
  - von ersetztem X
  - *zu jedem Symbol* von w
  - *eine separate Kante* nach unten

Wohlgeformte/korrekte Klammerausdrücke

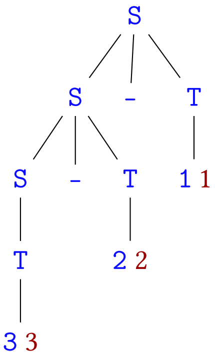
- $(\{X\}, \{(,)\}, X, \{X \to XX \mid (X) \mid \varepsilon\})$ erzeugt sogenannte *wohlgeformte* oder *korrekte Klammerausdrücke* (KKA).
- umgangssprachlich, das was sich ergibt,
  - wenn man in «normalen» arithmetischen Ausdrücken alles außer den Klammern weglässt
- (etwas) präziser
  - ε ist KKA.
  - 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



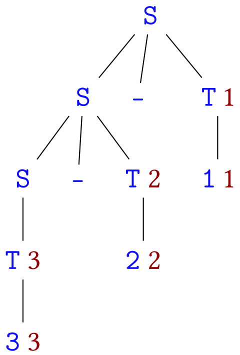
- $G = (N, T, \mathtt{S}, P)$ die Grammatik mit
  - Nichtterminalsymbolen $N = \{\mathtt{S}, \mathtt{T}\}$
  - Terminalsymbolen $T = \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{/}, \mathtt{(}, \mathtt{)}\}$
  - Startsymbol $\mathtt{S}$
  - Produktionen $P = \{\mathtt{S} \to \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
    $\mathtt{T} \to \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3} \mid \mathtt{5}\}.$
- betrachte `3-2-1`
  - Ableitungsbaum links
  - vergleiche `(3-2)-1` und `3-(2-1)`
- mehr in der Vorlesung «Sprachtechnologie und Compiler»

Arithmetische Ausdrücke



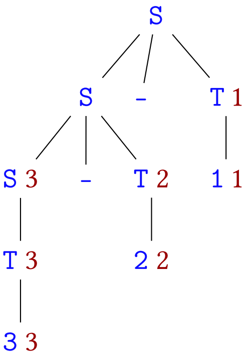
- $G = (N, T, \texttt{S}, P)$ die Grammatik mit
  - Nichtterminalsymbolen $N = \{\texttt{S}, \texttt{T}\}$
  - Terminalsymbolen $T = \{\texttt{1}, \texttt{2}, \texttt{3}, \texttt{+}, \texttt{-}, \texttt{*}, \texttt{/}, \texttt{(}, \texttt{)}\}$
  - Startsymbol S
  - Produktionen $P = \{\texttt{S} \to \texttt{T} \mid \texttt{S+T} \mid \texttt{S-T} \mid \texttt{S*T} \mid \texttt{S/T},$
  $\texttt{T} \to \texttt{(S)} \mid \texttt{1} \mid \texttt{2} \mid \texttt{3} \mid \texttt{5}\}.$
- betrachte 3-2-1
  - Ableitungsbaum links
  - vergleiche (3-2)-1 und 3-(2-1)
- mehr in der Vorlesung «Sprachtechnologie und Compiler»

Arithmetische Ausdrücke



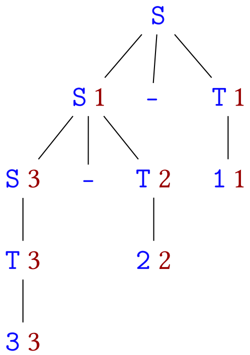
- $G = (N, T, \mathtt{S}, P)$ die Grammatik mit
  - Nichtterminalsymbolen $N = \{\mathtt{S}, \mathtt{T}\}$
  - Terminalsymbolen $T = \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{/}, \mathtt{(}, \mathtt{)}\}$
  - Startsymbol $\mathtt{S}$
  - Produktionen $P = \{\mathtt{S} \rightarrow \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
    $\mathtt{T} \rightarrow \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3} \mid \mathtt{5}\}.$
- betrachte `3-2-1`
  - Ableitungsbaum links
  - vergleiche `(3-2)-1` und `3-(2-1)`
- mehr in der Vorlesung «Sprachtechnologie und Compiler»

Arithmetische Ausdrücke



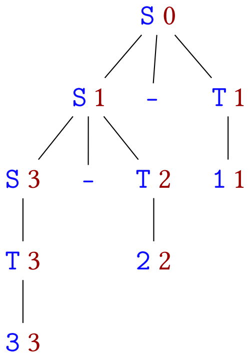
- $G = (N, T, \mathtt{S}, P)$ die Grammatik mit
  - Nichtterminalsymbolen $N = \{\mathtt{S}, \mathtt{T}\}$
  - Terminalsymbolen $T = \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{/}, \mathtt{(}, \mathtt{)}\}$
  - Startsymbol S
  - Produktionen $P = \{\mathtt{S} \to \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
    $\mathtt{T} \to \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3} \mid \mathtt{5}\}.$
- betrachte 3-2-1
  - Ableitungsbaum links
  - vergleiche (3-2)-1 und 3-(2-1)
- mehr in der Vorlesung «Sprachtechnologie und Compiler»

Arithmetische Ausdrücke



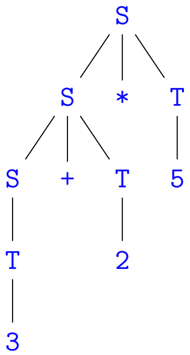
- $G = (N, T, \mathtt{S}, P)$ die Grammatik mit
  - Nichtterminalsymbolen $N = \{\mathtt{S}, \mathtt{T}\}$
  - Terminalsymbolen $T = \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{+}, \mathtt{-}, \mathtt{*}, \mathtt{/}, \mathtt{(}, \mathtt{)}\}$
  - Startsymbol S
  - Produktionen $P = \{\mathtt{S} \rightarrow \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
    $\mathtt{T} \rightarrow \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3} \mid \mathtt{5}\}.$
- betrachte 3-2-1
  - Ableitungsbaum links
  - vergleiche (3-2)-1 und 3-(2-1)
- mehr in der Vorlesung «Sprachtechnologie und Compiler»

Arithmetische Ausdrücke



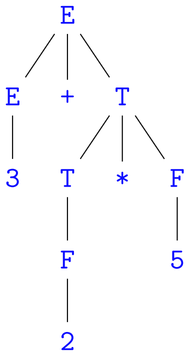
- $G = (N, T, \texttt{S}, P)$ die Grammatik mit
  - Nichtterminalsymbolen $N = \{\texttt{S}, \texttt{T}\}$
  - Terminalsymbolen $T = \{\texttt{1}, \texttt{2}, \texttt{3}, \texttt{+}, \texttt{-}, \texttt{*}, \texttt{/}, \texttt{(}, \texttt{)}\}$
  - Startsymbol S
  - Produktionen $P = \{\texttt{S} \rightarrow \texttt{T} \mid \texttt{S+T} \mid \texttt{S-T} \mid \texttt{S*T} \mid \texttt{S/T},$
    $\texttt{T} \rightarrow \texttt{(S)} \mid \texttt{1} \mid \texttt{2} \mid \texttt{3} \mid \texttt{5}\}.$
- betrachte 3-2-1
  - Ableitungsbaum links
  - vergleiche (3-2)-1 und 3-(2-1)
- mehr in der Vorlesung «Sprachtechnologie und Compiler»

Arithmetische Ausdrücke



- $G = (N, T, \mathtt{S}, P)$ 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 $P = \{\ \mathtt{S} \rightarrow \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
    $\mathtt{T} \rightarrow \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3} \mid \mathtt{5}\ \}.$
  - Diese Grammatik will man in der Praxis nicht!
  - betrachte Ableitungsbaum links oben zu `3+2*5`
  - vergleiche mit `(3+2)*5` und `3+(2*5)`

Arithmetische Ausdrücke



- $G = (N, T, \mathtt{S}, P)$ 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 $P = \{\ \mathtt{S} \longrightarrow \mathtt{T} \mid \mathtt{S+T} \mid \mathtt{S-T} \mid \mathtt{S*T} \mid \mathtt{S/T},$
    $\mathtt{T} \longrightarrow \mathtt{(S)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3} \mid \mathtt{5}\ \}.$
  - Diese Grammatik will man in der Praxis nicht!
  - betrachte Ableitungsbaum links oben zu $\mathtt{3+2*5}$
  - vergleiche mit $\mathtt{(3+2)*5}$ und $\mathtt{3+(2*5)}$
- bessere Idee: $G = (\{\mathtt{E}, \mathtt{T}, \mathtt{F}\}, T, \mathtt{E}, P)$ mit Term.sym. wie oben und
  - Produktionen $P = \{\ \mathtt{E} \longrightarrow \mathtt{T} \mid \mathtt{E+T} \mid \mathtt{E-T},$
    $\mathtt{T} \longrightarrow \mathtt{F} \mid \mathtt{T*F} \mid \mathtt{T/F},$
    $\mathtt{F} \longrightarrow \mathtt{(E)} \mid \mathtt{1} \mid \mathtt{2} \mid \mathtt{3}\ \}.$

Syntax aussagenlogischer Formeln

- leicht durch kontextfreie Grammatiken beschreibbar
- $G = (\{X\}, T, X, P)$ mit
- $T = Var_{AL} \cup \{(,), \neg, \wedge, \vee, \rightarrow\}$
  - $Var_{AL} \subseteq \{P_i \mid i \in \mathbb{N}_0\}$, endlich
- Produktionen auf der nächsten Folie
- Achtung: Verwechslungsgefahr bei Pfeilen
  - der Pfeil $\rightarrow$ in aussagenlogischen Formeln
  - der Pfeil $\longrightarrow$ 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

$$P = \{ X \longrightarrow \mathtt{P}_i \mid \mathtt{P}_i \in Var_{AL} \} \cup \{ X \longrightarrow (\neg X) \mid (X \wedge X) \mid (X \vee X) \mid (X \rightarrow X) \}$$

- Aus X sind genau die aussagenlogischen Formeln mit Aussagevariablen in Var_{AL} ableitbar.
- oder wie schon im Java-Beispiel:

"Backus-Naur-Form" BNF

$$\begin{aligned}\langle alF\rangle \longrightarrow\ & \mathtt{P}_i \qquad \text{für } \mathtt{P}_i \in Var_{AL} \\ & \mid (\neg \langle alF\rangle) \mid (\langle alF\rangle \wedge \langle alF\rangle) \\ & \mid (\langle alF\rangle \vee \langle alF\rangle) \mid (\langle alF\rangle \rightarrow \langle alF\rangle)\end{aligned}$$

Syntax aussagenlogischer Formeln — die Produktionen

- $G = (\{X\}, T, X, P)$ mit
  - $T = Var_{AL} \cup \{(,), \neg, \wedge, \vee, \rightarrow\}$
  - $Var_{AL} \subseteq \{P_i \mid i \in \mathbb{N}_0\}$, endlich
- Produktionenmenge

$$P = \{ X \longrightarrow P_i \mid P_i \in Var_{AL} \}$$
$$\cup \{ X \longrightarrow (\neg X) \mid (X \wedge X) \mid (X \vee X) \mid (X \rightarrow X) \}$$

- Aus X sind genau die aussagenlogischen Formeln mit Aussagevariablen in Var_{AL} ableitbar.
- oder wie schon im Java-Beispiel:

$$\begin{aligned} \langle alF \rangle ::= {} & P_i \qquad \text{für } P_i \in Var_{AL} \\ & \mid (\neg \langle alF \rangle) \mid (\langle alF \rangle \wedge \langle alF \rangle) \\ & \mid (\langle alF \rangle \vee \langle alF \rangle) \mid (\langle alF \rangle \rightarrow \langle alF \rangle) \end{aligned}$$

“Backus-Naur-Form” BNF

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 die erzeugende kontextfreie Grammatik konstruieren
  - zu kontextfreier Grammatik die erzeugte formale Sprache bestimmen

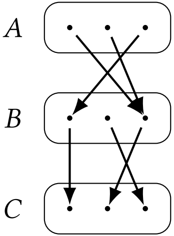
Wo sind wir?

Rekursive Definition syntaktischer Strukturen

Kontextfreie Grammatiken

Relationen (Teil 2)

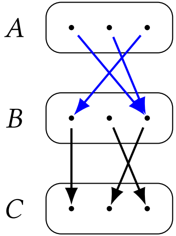
Erinnerung: Komposition von Funktionen



Erinnerung:

- Seien A, B, C Mengen,
- und $f : A \longrightarrow B$ und $g : B \longrightarrow C$,
- dann ist $g \circ f : A \longrightarrow C$ mit $(g \circ f)(x) = g(f(x))$.

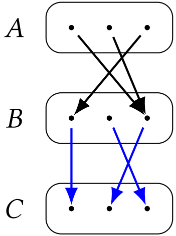
Erinnerung: Komposition von Funktionen



Erinnerung:

- Seien A, B, C Mengen,
- und $f : A \longrightarrow B$ und $g : B \longrightarrow C$,
- dann ist $g \circ f : A \longrightarrow C$ mit $(g \circ f)(x) = g(f(x))$.

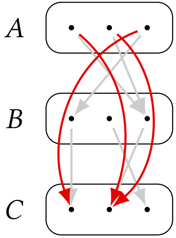
Erinnerung: Komposition von Funktionen



Erinnerung:

- Seien A, B, C Mengen,
- und $f : A \longrightarrow B$ und $g : B \longrightarrow C$,
- dann ist $g \circ f : A \longrightarrow C$ mit $(g \circ f)(x) = g(f(x))$.

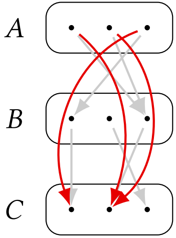
Erinnerung: Komposition von Funktionen



Erinnerung:

- Seien A, B, C Mengen,
- und $f : A \longrightarrow B$ und $g : B \longrightarrow C$,
- dann ist $g \circ f : A \longrightarrow C$ mit $(g \circ f)(x) = g(f(x))$.

Erinnerung: Komposition von Funktionen



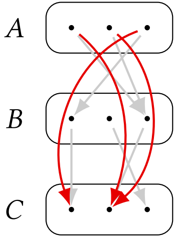
Erinnerung:

- Seien A, B, C Mengen,
- und $f : A \longrightarrow B$ und $g : B \longrightarrow C$,
- dann ist $g \circ f : A \longrightarrow C$ mit $(g \circ f)(x) = g(f(x))$.

Funktionen sind Relationen:

- $(g \circ f) \subseteq (A \times C)$
- Beobachtung: $(x, y) \in g \circ f \iff \exists \alpha \in B.(x, \alpha) \in f$ und $(\alpha, y) \in g$

Erinnerung: Komposition von Funktionen



Erinnerung:

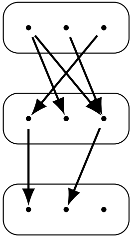
- Seien A, B, C Mengen,
- und $f : A \longrightarrow B$ und $g : B \longrightarrow C$,
- dann ist $g \circ f : A \longrightarrow C$ mit $(g \circ f)(x) = g(f(x))$.

Funktionen sind Relationen:

- $(g \circ f) \subseteq (A \times C)$
- Beobachtung: $(x, y) \in g \circ f \iff \exists \alpha \in B.(x, \alpha) \in f$ und $(\alpha, y) \in g$

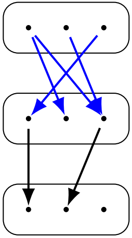
$\Longrightarrow$ Verallgemeinerung für allgemeine Relationen!

Komposition / Produkt von Relationen



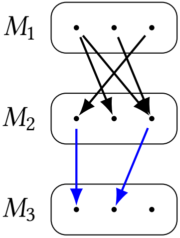
- Es seien $R \subseteq M_1 \times M_2$ und $S \subseteq M_2 \times M_3$ zwei Relationen
- *Produkt der Relationen* R und S
  - $S \circ R = \{(x, z) \in M_1 \times M_3 \mid$ es gibt $y \in M_2 : (x, y) \in R \wedge (y, z) \in S\}$
- *Infixschreibweise:* für alle $(x, z) \in M_1 \times M_3$
  - $x(S \circ R)z$ gdw. ein $y \in M_2$ existiert $: xRy \wedge ySz$
- $\circ$ ist *assoziativ:* falls auch noch $T \subseteq M_3 \times M_4$
  - $(T \circ S) \circ R = T \circ (S \circ R)$
- die identische Abbildung auf M als Relation I_M
  - $\mathrm{I}_M = \{(x, x) \mid x \in M\}$
- I_{M_i} neutrale Elemente bzgl. $\circ$
  - $R \circ \mathrm{I}_{M_1} = R = \mathrm{I}_{M_2} \circ R$

Komposition / Produkt von Relationen



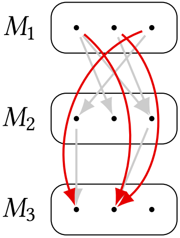
- Es seien $R \subseteq M_1 \times M_2$ und $S \subseteq M_2 \times M_3$ zwei Relationen
- *Produkt der Relationen* R und S
  - $S \circ R = \{(x, z) \in M_1 \times M_3 \mid$ es gibt $y \in M_2 : (x, y) \in R \wedge (y, z) \in S\}$
- *Infixschreibweise:* für alle $(x, z) \in M_1 \times M_3$
  - $x(S \circ R)z$ gdw. ein $y \in M_2$ existiert $: xRy \wedge ySz$
- $\circ$ ist *assoziativ:* falls auch noch $T \subseteq M_3 \times M_4$
  - $(T \circ S) \circ R = T \circ (S \circ R)$
- die identische Abbildung auf M als Relation I_M
  - $\mathrm{I}_M = \{(x, x) \mid x \in M\}$
- I_{M_i} neutrale Elemente bzgl. $\circ$
  - $R \circ \mathrm{I}_{M_1} = R = \mathrm{I}_{M_2} \circ R$

Komposition / 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. ein $y \in M_2$ existiert $: xRy \wedge ySz$
- $\circ$ ist *assoziativ:* falls auch noch $T \subseteq M_3 \times M_4$
  - $(T \circ S) \circ R = T \circ (S \circ R)$
- die identische Abbildung auf M als Relation I_M
  - $\mathrm{I}_M = \{(x, x) \mid x \in M\}$
- I_{M_i} neutrale Elemente bzgl. $\circ$
  - $R \circ \mathrm{I}_{M_1} = R = \mathrm{I}_{M_2} \circ R$

Komposition / 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. ein $y \in M_2$ existiert $: xRy \land ySz$
- $\circ$ ist *assoziativ:* falls auch noch $T \subseteq M_3 \times M_4$
  - $(T \circ S) \circ R = T \circ (S \circ R)$
- die identische Abbildung auf M als Relation I_M
  - $\mathrm{I}_M = \{(x,x) \mid x \in M\}$
- I_{M_i} 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

$u \Rightarrow^0 u$

$u \Rightarrow w \Rightarrow^i v$

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

  $$R^0 = \mathrm{I}_M$$

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

  Definition verträglich mit der von Funktionskomposition (!)

- 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
$$\begin{aligned} R \circ R &= \{(n, m) \mid \text{es gibt } k \in \mathbb{N}_0 : (n, k) \in R \text{ und } (k, m) \in R\} \\ &= \{(n, m) \mid \text{es gibt } k \in \mathbb{N}_0 : k = n+1 \text{ und } m = k+1 \in R\} \\ &= \{(n, n+2) \mid n \in \mathbb{N}_0\} \end{aligned}$$
- 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 $R \subseteq M \times M$ heißt *reflexiv*, wenn $I_M \subseteq R$, also wenn
  - für jedes $x \in M$: $x \; R \; x$
- Relation $R \subseteq M \times M$ 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
  - $\mathsf{I}_M = R^0 \subseteq R^*$
- 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 $R^* \subseteq S$, da jedes $R^i \subseteq S$
    - Kern des Induktionsbeweises:

Eigenschaften der reflexiv-transitiven Hülle — Erläuterungen

- R^* ist immer reflexiv
  - $I_M = R^0 \subseteq R^*$
- 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 $R^* \subseteq S$, da jedes $R^i \subseteq S$
    - Kern des Induktionsbeweises: $R^{i+1} = R^i \circ R \subseteq S \circ S \subseteq S$

Eigenschaften der reflexiv-transitiven Hülle — Erläuterungen

Karlsruher Institut für Technologie

- 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 $R^* \subseteq S$, da jedes $R^i \subseteq S$
    - Kern des Induktionsbeweises: $R^{i+1} = R^i \circ R \subseteq S \circ S \subseteq S$

Eigenschaften der reflexiv-transitiven Hülle — Erläuterungen

Karlsruher Institut für Technologie

- 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 $R^* \subseteq S$, da jedes $R^i \subseteq S$
    - Kern des Induktionsbeweises: $R^{i+1} = R^i \circ R \subseteq S \circ S \subseteq S$

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

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