

## 5. Modellierung von Strukturen

### 5.1 Kontextfreie Grammatiken

**Kontextfreie Grammatik (KFG):** formaler Kalkül, Ersetzungssystem; definiert

- **Sprache** als Menge von Sätzen; jeder **Satz** ist eine **Folge von Symbolen**
- **Menge von Bäumen;** jeder Baum repräsentiert die **Struktur eines Satzes** der Sprache

#### Anwendungen:

- Programme einer **Programmiersprache** und deren Struktur, z. B. Java, Pascal, C
- Sprachen als Schnittstellen zwischen Software-Werkzeugen, **Datenaustauschformate**, z. B. HTML, XML
- Bäume zur Repräsentation **strukturierter Daten**, z. B. in HTML
- Struktur von **Protokollen** beim Austausch von Nachrichten zwischen Geräten oder Prozessen

#### Beispiel zu HTML:

```
<table>
  <tr>
    <td>Mo</td>
    <td>11-13</td>
    <td>AM</td>
  </tr>
  <tr>
    <td>Fr</td>
    <td>9-11</td>
    <td>AM</td>
  </tr>
</table>
```

### Vorlesung Modellierung WS 2001/2002 / Folie 501

#### Ziele:

Einsatz von KFGn kennenlernen

#### in der Vorlesung:

Erläuterungen zu den Anwendungen

#### nachlesen:

G. Goos: Vorl. über Informatik Bd.1, Abschnitt 1.6.3

# Kontextfreie Grammatik

Eine kontextfreie Grammatik  $G = (T, N, P, S)$  besteht aus:

$T$	<b>Menge der Terminalsymbole</b> (kurz: Terminale)
$N$	<b>Menge der Nichtterminalsymbole</b> (kurz: Nichtterminale)
	$T$ und $N$ sind disjunkte Mengen
$S \in N$	<b>Startsymbol</b> (auch Zielsymbol)
$P \subseteq N \times V^*$	<b>Menge der Produktionen</b> ; $(A, x) \in P$ , mit $A \in N$ und $x \in V^*$ ; statt $(A, x)$ schreibt man $A ::= x$
$V = T \cup N$	heißt auch <b>Vokabular</b> , seine Elemente heißen <b>Symbole</b>

Man sagt „In der Produktion  $A ::= x$  steht  $A$  auf der **linken Seite** und  $x$  auf der **rechten Seite**.“

Man gibt Produktionen häufig **Namen**:  $p1: A ::= x$

In Symbolfolgen aus  $V^*$  werden die Elemente nur durch Zwischenraum getrennt:  $A ::= B C D$

**Beispiel:**

	Name	Produktionsmenge P
<b>Terminale</b>	$T = \{ (, ) \}$	$N \quad V^*$
<b>Nichtterminale</b>	$N = \{ \text{Klammern, Liste} \}$	{
<b>Startsymbol</b>	$S = \text{Klammern}$	$p1: \text{Klammern} ::= '( \text{ Liste } )'$
		$p2: \text{Liste} ::= \text{Klammern Liste}$
		$p3: \text{Liste} ::=$
		}

## Vorlesung Modellierung WS 2001/2002 / Folie 502

**Ziele:**

KFG Definition lernen

**in der Vorlesung:**

- Erläuterung der Begriffe an dem Beispiel
- Erläuterung der Notation von Produktionen
- Unbenannte Terminale werden gekennzeichnet, um Verwechslungen mit KFG-Zeichen zu vermeiden: '('

**nachlesen:**

G. Goos: Vorl. über Informatik Bd.1, Abschnitt 1.6.3

# Bedeutung der Produktionen

Eine Produktion  $A ::= x$  ist eine **Strukturregel**: A besteht aus x

## Beispiele:

DeutscherSatz ::= Subjekt Prädikat Objekt  
*Ein*DeutscherSatz *besteht aus (der Folge)* Subjekt Prädikat Objekt

Klammern ::= '(' Liste ')'

Zuweisung ::= Variable '=' Ausdruck  
 Variable ::= Variable '[' Ausdruck ']'

**Produktion graphisch als gewurzelter Baum** mit geordneten Kanten und mit Symbolen als Knotenmarken:



## Vorlesung Modellierung WS 2001/2002 / Folie 503

### Ziele:

Produktionen verstehen

### in der Vorlesung:

Erläuterungen der beiden Rollen von Produktionen:

- Definition von Struktur: "besteht aus"
- Definition von Ersetzungen
- Siehe auch Mod-5.5 zur graphischen Darstellung von Produktionen.

### nachlesen:

G. Goos: Vorl. über Informatik Bd.1, Abschnitt 1.6.3

# Ableitungen

Produktionen sind **Ersetzungsregeln**: Ein Nichtterminal  $A$  in einer Symbolfolge  $u A v$  kann durch die rechte Seite  $x$  einer Produktion  $A ::= x$  ersetzt werden.

Das ist ein **Ableitungsschritt**; er wird notiert als  $u A v \Rightarrow u x v$

z. B. **Klammern Klammern Liste**  $\Rightarrow$  **Klammern ( Liste ) Liste**  
mit Produktion  $p_1$

Beliebig viele Ableitungsschritte nacheinander angewandt heißen **Ableitung**; notiert als  $u \Rightarrow^* v$

Eine kontextfreie Grammatik **definiert eine Sprache**; das ist eine Menge von Sätzen.

Jeder Satz ist eine Folge von Terminalsymbolen, die aus dem Startsymbol ableitbar ist:

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

Grammatik auf Mod-5.2 definiert geschachtelte Folgen paariger Klammern als Sprachmenge:

$$\{ (), (()), (())(), ((())()), \dots \} \subseteq L(G)$$

Ableitung des Satzes  $((())())$ :

<b>S</b>	<b>= Klammern</b>
	$\Rightarrow ( \text{Liste} )$
	$\Rightarrow ( \text{Klammern Liste} )$
	$\Rightarrow ( \text{Klammern Klammern Liste} )$
	$\Rightarrow ( \text{Klammern ( Liste ) Liste} )$
	$\Rightarrow ( ( \text{Liste} ) ( \text{Liste} ) \text{Liste} )$
	$\Rightarrow ( () ( \text{Liste} ) \text{Liste} )$
	$\Rightarrow ( () () \text{Liste} )$
	$\Rightarrow ( () () )$

## Vorlesung Modellierung WS 2001/2002 / Folie 504

### Ziele:

Ableitungsbegriff verstehen

### in der Vorlesung:

Erläuterungen dazu

- Beispiele für Ableitungen
- Beispiele für Sprachen

### nachlesen:

G. Goos: Vorl. über Informatik Bd.1, Abschnitt 1.6.3

# Ableitungsbäume

Jede Ableitung kann man als **gewurzelten Baum** darstellen:

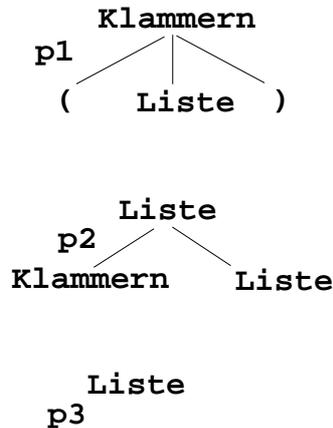
Die **Knoten** mit ihren Marken repräsentieren **Vorkommen von Symbolen**.

Ein Knoten mit seinen direkten Nachbarn repräsentiert die **Anwendung einer Produktion**.

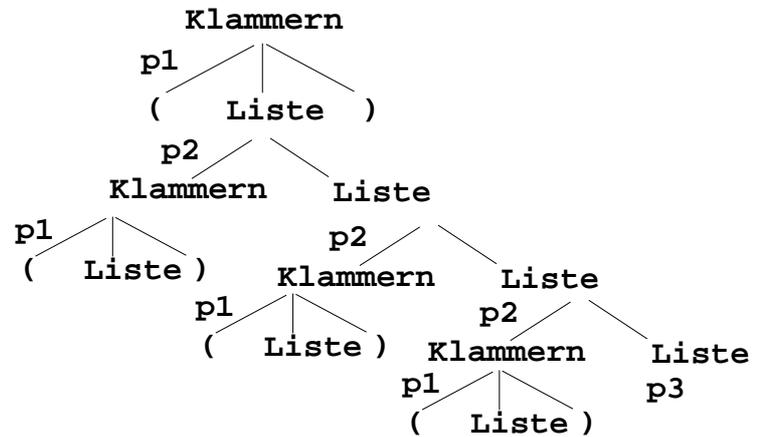
Die **Wurzel** ist mit dem **Startsymbol** markiert.

**Terminale** kommen nur an **Blättern** vor.

Produktionen:



ein Ableitungsbaum:



der Satz dazu:  $((()()))$

Satz zum Baum: Terminale im links-abwärts Durchgang

## Vorlesung Modellierung WS 2001/2002 / Folie 505

### Ziele:

Ableitungsbaum verstehen

### in der Vorlesung:

- Konstruktion des Baumes durch Zusammensetzen von Produktionsanwendungen,
- Zusammenhang zum Satz der Sprache

### nachlesen:

G. Goos: Vorl. über Informatik Bd.1, Abschnitt 1.6.3

## Beispiel: Ausdrucksgrammatik

p1: Ausdruck ::= Ausdruck BinOpr Ausdruck

p2: Ausdruck ::= Zahl

p3: Ausdruck ::= Bezeichner

p4: Ausdruck ::= '(' Ausdruck ')'

p5: BinOpr ::= '+'

p6: BinOpr ::= '-'

p7: BinOpr ::= '\*'

p8: BinOpr ::= '/'

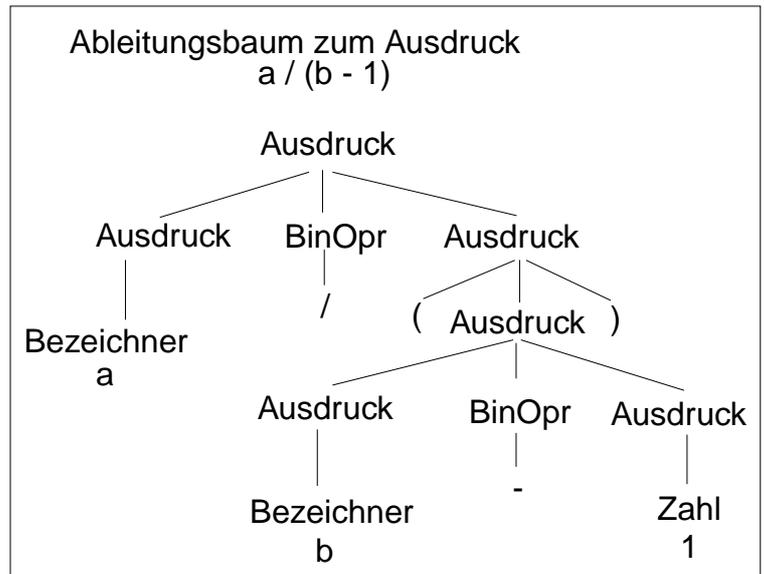
Startsymbol: Ausdruck

Terminale:

{ Zahl, Bezeichner, (, ), +, -, \*, / }

Schreibweise der Terminale

Zahl und Bezeichner wird nicht in der KFG definiert.



**Grammatik ist mehrdeutig:** Es gibt **Sätze**, die mehrere **Ableitungsbäume** haben.

## Vorlesung Modellierung WS 2001/2002 / Folie 506

### Ziele:

Vollständiges Beispiel sehen

### in der Vorlesung:

- Erläuterungen dazu.
- Vergleich mit Kantorowitsch-Bäumen.
- Diese Grammatik ist mehrdeutig: z. B. hat der Satz  $a+b+c$  mehrere Ableitungsbäume.

### nachlesen:

G. Goos: Vorl. über Informatik Bd.1, Abschnitt 1.6.3

## Beispiel: Tabellen in HTML

**HTML:** Hypertext Markup Language zur Darstellung von verzeigerten Dokumenten, insbesondere im WWW verwendet.

**typisch: geklammerte Strukturen** mit Klammern der Form `<x>...</x>`.

hier: vereinfachter Ausschnitt aus der Sprache zur Darstellung von Tabellen.

### Produktionen der kontextfreien Grammatik:

```
Table ::= '<table>' Rows '</table>'
Rows  ::= Row *
Row   ::= '<tr>' Cells '</tr>'
Cells ::= Cell *
Cell  ::= '<td>' Text '</td>'
Cell  ::= '<td>' Table '</td>'
```

### Beispieltext in HTML:

```
<table>
  <tr> <td>Tag</td>
      <td>Zeit</td>
      <td>Raum</td></tr>
  <tr> <td>Mo</td>
      <td>11:00-12.30</td>
      <td>AM</td></tr>
  <tr> <td>Fr</td>
      <td>9:15-10:45</td>
      <td>AM</td></tr>
</table>
```

### Erweiterung der Notation vonKFGn:

**X \*** auf der rechten Seite einer Produktion steht für eine **beliebig lange Folge von X**

(gleiche Bedeutung wie bei Wertebereichen)

### Darstellung der Tabelle:

Tag	Zeit	Raum
Mo	11:00-12.30	AM
Fr	9:15-10:45	AM

## Vorlesung Modellierung WS 2001/2002 / Folie 507

### Ziele:

HTML-Ausschnitt verstehen

### in der Vorlesung:

Erläuterungen

- zum \*-Operator (siehe Mod-2.8a),
- zur Struktur von HTML,
- zum Beispiel,
- zur Baumdarstellung

### Übungsaufgaben:

Beschreiben Sie die Operationsfolgen zur Bedienung des Getränkeautomaten durch eine KFG.