

# Theoretische Grundlagen der Informatik

Vorlesung am 10.01.2017

INSTITUT FÜR THEORETISCHE INFORMATIK



**Syntaxbäume** visualisieren die Ableitung eines einzelnen Wortes.

- An der Wurzel eines Syntaxbaumes steht das Startsymbol.
- Jeder innere Knoten enthält eine Variable.
- Die Blätter sind Symbole aus  $\Sigma$  oder  $\varepsilon$ .
- Wenn ein innerer Knoten  $A$  als Nachfolger von links nach rechts  $\alpha_1, \dots, \alpha_r \in V \cup \Sigma$  hat, so muss  $A \rightarrow \alpha_1 \dots \alpha_r$  eine Ableitungsregel der Grammatik sein.

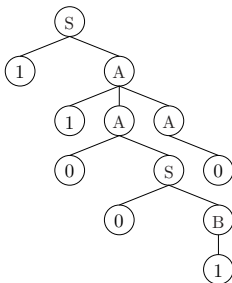
# Syntaxbäume - Beispiel

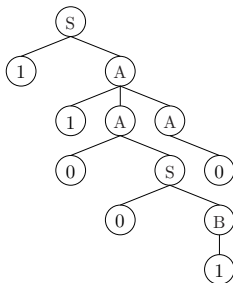
Zu den Regeln

$$R = \{S \rightarrow 0B \mid 1A, A \rightarrow 0 \mid 0S \mid 1AA, B \rightarrow 1 \mid 1S \mid 0BB\}$$

betrachte die Ableitung

$$S \rightarrow 1A \rightarrow 11AA \rightarrow 11A0 \rightarrow 110S0 \rightarrow 1100B0 \rightarrow 110010.$$





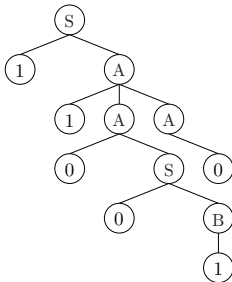
- Zu jeder Ableitung gehört genau ein Syntaxbaum.
- Zu jedem Syntaxbaum gehören jedoch verschiedene Ableitungen des gleichen Wortes.

# Syntaxbäume - Beispiel

$$R = \{S \rightarrow 0B \mid 1A, A \rightarrow 0 \mid 0S \mid 1AA, B \rightarrow 1 \mid 1S \mid 0BB\}$$

$S \rightarrow 1A \rightarrow 11AA \rightarrow 110SA \rightarrow 1100BA \rightarrow 11001A \rightarrow 110010$

$S \rightarrow 1A \rightarrow 11AA \rightarrow 11A0 \rightarrow 110S0 \rightarrow 1100B0 \rightarrow 110010$



Für Chomsky-2 Grammatiken gilt:

- Wegen der Kontextfreiheit ist die Reihenfolge, in der abgeleitet wird, für das Ergebnis unerheblich.

Eine **Linksableitung** (**Rechtsableitung**) ist eine Ableitung, bei der in jedem Schritt die linkeste (rechtste) Variable abgeleitet wird.

Eine kontextfreie Grammatik  $G$  heißt **eindeutig**, wenn es für jedes Wort  $w \in L(G)$  genau einen Syntaxbaum gibt.

Eine kontextfreie Sprache  $L$  heißt **eindeutig**, wenn es eine eindeutige Grammatik  $G$  mit  $L(G) = L$  gibt. Ansonsten heißt  $L$  **inhärent mehrdeutig**.

# Beispiel

Die Sprache

$$L = \{0^n 1^n \mid n \geq 1\}$$

erzeugt durch die Grammatik

$$V = \{S\}$$

$$\Sigma = \{0, 1\}$$

$$R = \{S \rightarrow 01 \mid 0S1\}$$

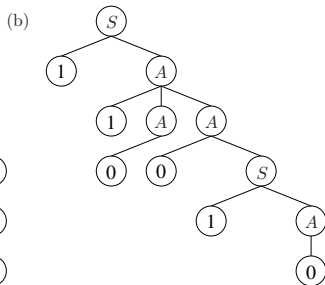
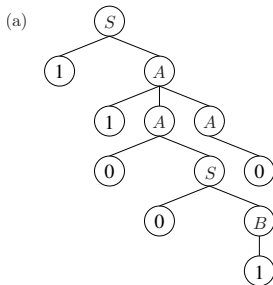
ist eindeutig.

# Beispiel

Die Sprache gegeben durch die Regeln

$$R = \{S \rightarrow 0B \mid 1A, A \rightarrow 0 \mid 0S \mid 1AA, B \rightarrow 1 \mid 1S \mid 0BB\}$$

ist nicht eindeutig.





Eine kontextfreie Grammatik ist in **Chomsky-Normalform**, wenn alle Regeln von der Form:

$$A \rightarrow BC \quad \text{oder} \quad A \rightarrow a$$

sind, mit  $A, B, C \in V$  und  $a \in \Sigma$ .

Eine kontextfreie Grammatik ist in **Chomsky-Normalform**, wenn alle Regeln von der Form:

$$A \rightarrow BC \quad \text{oder} \quad A \rightarrow a$$

sind, mit  $A, B, C \in V$  und  $a \in \Sigma$ .

- Grammatiken in Chomsky-Normalform können also nicht das Wort  $\varepsilon$  erzeugen.
- Für kontextfreie Sprachen, die  $\varepsilon$  enthalten, lässt sich eine Grammatik in Chomsky-Normalform leicht erweitern durch die Regeln

$$S' \rightarrow \varepsilon \quad \text{und} \quad S' \rightarrow S,$$

wobei  $S'$  ein neues Startsymbol zur Erzeugung von  $\varepsilon$  ist.

## Satz:

Jede kontextfreie Grammatik kann in eine Grammatik in erweiterter Chomsky–Normalform überführt werden.

Eine Grammatik in erweiterter Chomsky–Normalform ist eine Grammatik in Chomsky–Normalform mit den zusätzlichen Regeln  $S' \rightarrow \varepsilon$  und  $S' \rightarrow S$ .

## Beweis (konstruktiv):

- Wir geben eine Schritt–für–Schritt–Überführung der Regeln in Regeln in Normalform an.
- Großbuchstaben repräsentieren immer Nichtterminale.
- Kleinbuchstaben repräsentieren immer Terminale.

Wir veranschaulichen den Beweis an folgendem Beispiel:  
Grammatik  $G = (\Sigma, V, S, R)$  mit

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

$$V = \{A, B, C, D, E, S\}$$

und der folgenden Regelmenge  $R$ :

$$S \rightarrow A \mid aAa \mid bBb \mid \varepsilon$$

$$A \rightarrow C \mid a$$

$$B \rightarrow b$$

$$C \rightarrow AF \mid CDE \mid \varepsilon$$

$$D \rightarrow A \mid B \mid ab$$

$$E \rightarrow B$$

$$F \rightarrow D \mid E$$

## Ziel:

- Alle Regeln enthalten auf der rechten Seite nur Symbole aus  $V$  oder nur ein Symbol aus  $\Sigma$ .

## Vorgehen:

- Ersetze dazu in allen rechten Seiten von Regeln Symbole  $a \in \Sigma$  durch neue Variablen  $Y_a$  und füge die Regeln  $Y_a \rightarrow a$  hinzu.

## Ziel:

- Alle Regeln enthalten auf der rechten Seite nur Symbole aus  $V$  oder nur ein Symbol aus  $\Sigma$ .

## Vorgehen:

- Ersetze dazu in allen rechten Seiten von Regeln Symbole  $a \in \Sigma$  durch neue Variablen  $Y_a$  und füge die Regeln  $Y_a \rightarrow a$  hinzu.

$$S \rightarrow A \mid aAa \mid bBb \mid \varepsilon$$

$$A \rightarrow C \mid a$$

$$B \rightarrow b$$

$$C \rightarrow AF \mid CDE \mid \varepsilon$$

$$D \rightarrow A \mid B \mid ab$$

$$E \rightarrow B$$

$$F \rightarrow D \mid E$$

 $\Rightarrow$ 

$$S \rightarrow A \mid Y_aAY_a \mid Y_bBY_b \mid \varepsilon$$

$$A \rightarrow C \mid a$$

$$B \rightarrow b$$

$$C \rightarrow AF \mid CDE \mid \varepsilon$$

$$D \rightarrow A \mid B \mid Y_aY_b$$

$$E \rightarrow B$$

$$F \rightarrow D \mid E$$

$$Y_a \rightarrow a$$

$$Y_b \rightarrow b$$

## Schritt 2

### Ziel:

- Alle rechten Seiten haben Länge  $\leq 2$ .

### Vorgehen:

- Sei  $A \rightarrow B_1 \dots B_m$  Regel mit  $m > 2$ .
- Führe  $m - 2$  neue Variablen  $C_1, \dots, C_{m-2}$  ein, und ersetze die Regel

$$A \rightarrow B_1 \dots B_m$$

durch neue Regeln

$$\begin{aligned} A &\rightarrow B_1 C_1 \\ C_i &\rightarrow B_{i+1} C_{i+1} \quad \text{für } 1 \leq i \leq m-3 \\ C_{m-2} &\rightarrow B_{m-1} B_m \end{aligned}$$

### Ziel:

- Alle rechten Seiten haben Länge  $\leq 2$ .

$$S \rightarrow A \mid Y_a A Y_a \mid Y_b B Y_b \mid \varepsilon$$

$$A \rightarrow C \mid a$$

$$B \rightarrow b$$

$$C \rightarrow AF \mid CDE \mid \varepsilon$$

$$D \rightarrow A \mid B \mid Y_a Y_b$$

$$E \rightarrow B$$

$$F \rightarrow D \mid E$$

$\Rightarrow$

$$S \rightarrow A \mid Y_a C_1 \mid Y_b C_2 \mid \varepsilon$$

$$A \rightarrow C \mid a$$

$$B \rightarrow b$$

$$C \rightarrow AF \mid C C_3 \mid \varepsilon$$

$$D \rightarrow A \mid B \mid Y_a Y_b$$

$$E \rightarrow B$$

$$F \rightarrow D \mid E$$

$$C_1 \rightarrow A Y_a$$

$$C_2 \rightarrow B Y_b$$

$$C_3 \rightarrow DE$$

$$Y_a \rightarrow a$$

$$Y_b \rightarrow b$$

$$Y_a \rightarrow a$$

$$Y_b \rightarrow b$$



### Ziel:

- Es kommen keine Regeln  $A \rightarrow \varepsilon$  vor.

### Vorgehen, Phase 1:

**Finde die Menge  $V'$  aller Variablen  $A$  für die  $A \xrightarrow{*} \varepsilon$  existiert:**

- Es werden erst alle  $A$  mit  $A \rightarrow \varepsilon$  in  $V'$  aufgenommen.
- Dann wird geprüft, ob neue Regeln  $B \rightarrow \varepsilon$  entstehen, wenn man  $A$  in allen Regeln auf der rechten Seite  $A$  durch  $\varepsilon$  ersetzt.
- Ist dies der Fall, so werden die entsprechenden Variablen  $B$  in  $V'$  aufgenommen und genauso behandelt.
- Das Verfahren hört auf, wenn  $V'$  sich nicht mehr ändert.

### Ziel:

- Es kommen keine Regeln  $A \rightarrow \varepsilon$  vor.

### Vorgehen, Phase 1:

**Finde die Menge  $V'$  aller Variablen  $A$  für die  $A \xrightarrow{*} \varepsilon$  existiert:**

- Bemerkung: In Phase 1 werden noch keine Regeln geändert.
- Die Ersetzung ist also nur „testweise“.
- Am Ende enthält  $V'$  alle Variablen  $A$  mit  $A \xrightarrow{*} \varepsilon$ .

### Ziel:

- Es kommen keine Regeln  $A \rightarrow \varepsilon$  vor.

### Vorgehen, Phase 2: Ersetzung.

- Gegeben  $V'$  aus Phase 1.
  - Streiche alle Regeln  $A \rightarrow \varepsilon$ .
  - Für  $A \rightarrow BC$  füge die zusätzliche Regel
    - $A \rightarrow B$  falls  $C \in V'$ ,
    - $A \rightarrow C$  falls  $B \in V'$
- ein.
- (Die Regel  $A \rightarrow BC$  wird nicht gestrichen).

- Initialisierung  
 $V' = \{A \mid A \rightarrow \varepsilon\} = \{S, C\}$
- Erster Durchlauf liefert:  $A$   
 $\Rightarrow V' = \{A, C, S\}$
- Zweiter Durchlauf liefert:  $D$   
 $\Rightarrow V' = \{A, C, D, S\}$
- Dritter Durchlauf liefert:  $F$   
 $\Rightarrow V' = \{A, C, D, F, S\}$
- Vierter Durchlauf liefert nichts neues

$$\begin{aligned} S &\rightarrow A \mid Y_a C_1 \mid Y_b C_2 \mid \varepsilon \\ A &\rightarrow C \mid a \\ B &\rightarrow b \\ C &\rightarrow AF \mid CC_3 \mid \varepsilon \\ D &\rightarrow A \mid B \mid Y_a Y_b \\ E &\rightarrow B \\ F &\rightarrow D \mid E \\ C_1 &\rightarrow AY_a \\ C_2 &\rightarrow BY_b \\ C_3 &\rightarrow DE \\ Y_a &\rightarrow a \\ Y_b &\rightarrow b \end{aligned}$$

## Schritt 3

- Streiche  $\varepsilon$ -Produktionen
- Simuliere Ableitungen  $A \xrightarrow{*} \varepsilon$  auf den verbleibenden Regeln  
 $V' = \{A, C, D, F, S\}$

$$S \rightarrow A \mid Y_a C_1 \mid Y_b C_2$$

$$A \rightarrow C \mid a$$

$$B \rightarrow b$$

$$C \rightarrow AF \mid F \mid A \mid CC_3 \mid C_3$$

$$D \rightarrow A \mid B \mid Y_a Y_b$$

$$E \rightarrow B$$

$$F \rightarrow D \mid E$$

$$C_1 \rightarrow AY_a \mid Y_a$$

$$C_2 \rightarrow BY_b$$

$$C_3 \rightarrow DE \mid E$$

$$Y_a \rightarrow a$$

$$Y_b \rightarrow b$$

## Ziel

- Die Grammatik enthält keine (Ketten-)Regeln der Form  $A \rightarrow B$ .
- Beispiel

$$A \rightarrow B \mid C$$

$$B \rightarrow C$$

$$C \rightarrow c$$

## Ziel

- Die Grammatik enthält keine (Ketten-)Regeln der Form  $A \rightarrow B$ .

## Vorgehen

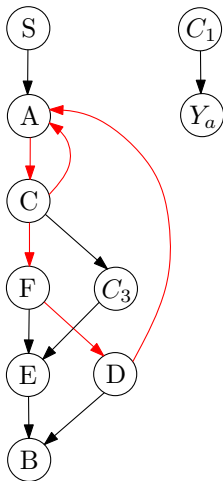
- **Phase 1:** Finde alle Kreise  $A_1 \rightarrow A_2 \rightarrow \dots \rightarrow A_n \rightarrow A_1$ .  
Ersetze alle  $A_j$  durch  $A_1$ .
- **Phase 2:** Betrachte die Regeln der Form  $A \rightarrow B$  in umgekehrt topologischer Reihenfolge.
  - Für Regel  $A \rightarrow B$  und jede Regel  $B \rightarrow b$  füge Regel  $A \rightarrow b$  hinzu.
  - Lösche Regel  $A \rightarrow B$ .

## Topologische Sortierung der Regelmenge

- $V_1, \dots, V_k$  Menge von Variablen aus Kettenregeln
- $V_1, \dots, V_k$  topologisch sortiert, wenn gilt:
- $V_i \xrightarrow{*} V_j \Rightarrow i < j$
- Voraussetzung: Es gibt keine zyklischen Abhängigkeiten



# Abhängigkeitsgraph



$$S \rightarrow A \mid Y_a C_1 \mid Y_b C_2$$

$$A \rightarrow C \mid a$$

$$B \rightarrow b$$

$$C \rightarrow AF \mid F \mid A \mid CC_3 \mid C_3$$

$$D \rightarrow A \mid B \mid Y_a Y_b$$

$$E \rightarrow B$$

$$F \rightarrow D \mid E$$

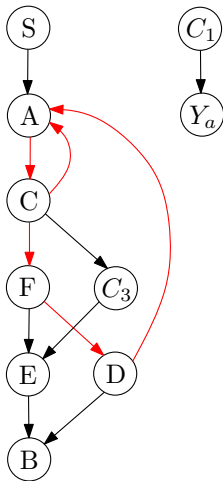
$$C_1 \rightarrow AY_a \mid Y_a$$

$$C_2 \rightarrow BY_b$$

$$C_3 \rightarrow DE \mid E$$

$$Y_a \rightarrow a$$

$$Y_b \rightarrow b$$



- Zyklen  $A \rightarrow C$  und  $A \rightarrow C \rightarrow F \rightarrow D \rightarrow A$
- $\Rightarrow A, C, F, D$  äquivalent
- Entferne an Zyklen beteiligte Regeln
- Ersetze Vorkommen von  $C, F, D$  in allen Regeln durch  $A$
- Lösche Regeln der Form  $A \rightarrow A$

# Schritt 4 – Phase 1

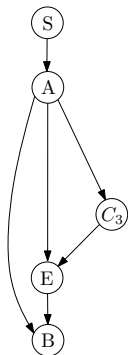
Zyklus:  $A \rightarrow C \rightarrow F \rightarrow D \rightarrow A$

$S \rightarrow A \mid Y_a C_1 \mid Y_b C_2$   
 $A \rightarrow C \mid a$   
 $B \rightarrow b$   
 $C \rightarrow AF \mid F \mid A \mid C C_3 \mid C_3$   
 $D \rightarrow A \mid B \mid Y_a Y_b$   
 $E \rightarrow B$   
 $F \rightarrow D \mid E$   
 $C_1 \rightarrow A Y_a \mid Y_a$   
 $C_2 \rightarrow B Y_b$   
 $C_3 \rightarrow D E \mid E$   
 $Y_a \rightarrow a$   
 $Y_b \rightarrow b$

$\Rightarrow$

$S \rightarrow A \mid Y_a C_1 \mid Y_b C_2$   
 $A \rightarrow a$   
 $B \rightarrow b$   
 $A \rightarrow AA \mid A C_3 \mid C_3$   
 $A \rightarrow B \mid Y_a Y_b$   
 $E \rightarrow B$   
 $A \rightarrow E$   
 $C_1 \rightarrow A Y_a \mid Y_a$   
 $C_2 \rightarrow B Y_b$   
 $C_3 \rightarrow A E \mid E$   
 $Y_a \rightarrow a$   
 $Y_b \rightarrow b$

## Schritt 4 - Phase 2



$$S \rightarrow A \mid Y_a C_1 \mid Y_b C_2$$

$$A \rightarrow a \mid AA \mid AC_3 \mid C_3 \mid B \mid Y_a Y_b \mid E$$

$$B \rightarrow b$$

$$E \rightarrow B$$

$$C_1 \rightarrow AY_a \mid Y_a$$

$$C_2 \rightarrow BY_b$$

$$C_3 \rightarrow AE \mid E$$

$$Y_a \rightarrow a$$

$$Y_b \rightarrow b$$

Topologische Sortierung:  $S, A, C_3, E, B, C_1, Y_a$

## Schritt 4 - Phase 2

- Gehe in umgekehrter topologischer Sortierung vor
- Ersetze  $A \rightarrow B$  durch  $A \rightarrow \beta$ , falls Regel  $B \rightarrow \beta$  existiert
- Topologische Sortierung:  $S, A, C_3, E, B, C_1, Y_a$

$$\begin{array}{l} S \rightarrow A \mid Y_a C_1 \mid Y_b C_2 \\ A \rightarrow a \mid AA \mid AC_3 \mid C_3 \mid \\ \quad B \mid Y_a Y_b \mid E \\ B \rightarrow b \\ E \rightarrow B \\ C_1 \rightarrow AY_a \mid Y_a \\ C_2 \rightarrow BY_b \\ C_3 \rightarrow AE \mid E \\ Y_a \rightarrow a \\ Y_b \rightarrow b \end{array} \quad \Rightarrow \quad \begin{array}{l} S \rightarrow Y_a C_1 \mid Y_b C_2 \mid a \mid AA \\ \quad AC_3 \mid AE \mid b \mid Y_a Y_b \\ A \rightarrow a \mid AA \mid AC_3 \mid AE \mid \\ \quad b \mid Y_a Y_b \\ B \rightarrow b \\ E \rightarrow b \\ C_1 \rightarrow AY_a \mid a \\ C_2 \rightarrow BY_b \\ C_3 \rightarrow AE \mid b \\ Y_a \rightarrow a \\ Y_b \rightarrow b \end{array}$$

## Sonderbehandlung von $\varepsilon$ -Produktionen

- Grammatik ist nun in Chomsky-Normalform, aber
- $G$  enthält Ableitungsfolge  $S \xrightarrow{*} \varepsilon$  (haben wir entfernt).
- Erweitere Grammatik um Regeln  $S' \rightarrow S$  und  $S' \rightarrow \varepsilon$  für neues Startsymbol  $S'$ .

**Satz:**

Es gibt einen Algorithmus (den Cocke–Younger–Kasami Algorithmus), der für eine kontextfreie Grammatik  $G$  in Chomsky–Normalform und ein Wort  $w \in \Sigma^*$  in Zeit  $\mathcal{O}(|R| \cdot n^3)$  entscheidet, ob  $w \in L(G)$ , wobei  $n = |w|$  und  $|R|$  die Anzahl der Regeln von  $G$  ist.

- Sei  $w = w_1 \dots w_n$ .
- Sei  $V_{ij} \subseteq V$  so dass  $A \xrightarrow{*} w_i \dots w_j$  impliziert  $A \in V_{ij}$ .
- Für alle  $1 \leq i \leq j \leq n$  berechne die Menge  $V_{ij} \subseteq V$ .
- Dann ist  $w \in L(G)$  genau dann, wenn  $S \in V_{1n}$  ist.
  
- Die Tabelle der  $V_{ij}$  wird nach wachsendem  $\ell := j - i$  aufgebaut, beginnend mit  $\ell = 0$ .
- Für  $j - i = \ell > 0$  wird die Berechnung von  $V_{ij}$  systematisch auf zuvor berechnete  $V_{ik}, V_{k+1j}$  mit  $i \leq k < j$  zurückgeführt.

## Bemerkung

- Wir benutzen dynamische Programmierung.



# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

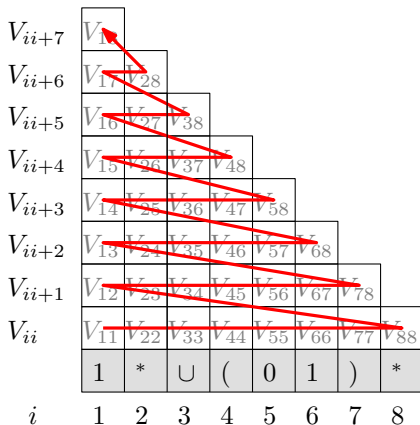
$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$

$V_{ii+7}$	$V_{18}$							
$V_{ii+6}$	$V_{17}$	$V_{28}$						
$V_{ii+5}$	$V_{16}$	$V_{27}$	$V_{38}$					
$V_{ii+4}$	$V_{15}$	$V_{26}$	$V_{37}$	$V_{48}$				
$V_{ii+3}$	$V_{14}$	$V_{25}$	$V_{36}$	$V_{47}$	$V_{58}$			
$V_{ii+2}$	$V_{13}$	$V_{24}$	$V_{35}$	$V_{46}$	$V_{57}$	$V_{68}$		
$V_{ii+1}$	$V_{12}$	$V_{23}$	$V_{34}$	$V_{45}$	$V_{56}$	$V_{67}$	$V_{78}$	
$V_{ii}$	$V_{11}$	$V_{22}$	$V_{33}$	$V_{44}$	$V_{55}$	$V_{66}$	$V_{77}$	$V_{88}$
$i$	1	2	3	4	5	6	7	8

# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y_{\cup} &\rightarrow \cup \\
 Y_{(} &\rightarrow ( \\
 Y_{)} &\rightarrow )
 \end{aligned}$$



**Fall  $\ell = 0$ :**

- Konstruiere die Mengen  $V_{ij}$ , d.h. alle  $A \in V$  mit  $A \xrightarrow{*} w_i$ .
- Da  $G$  in Chomsky–Normalform ist, gilt  $A \xrightarrow{*} w_i$  nur, wenn  $(A \rightarrow w_i) \in R$ .
- Die Berechnung von  $V_{ij}$  ist für alle  $i \in \{1, \dots, n\}$  in  $\mathcal{O}(|R|)$  möglich.

# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y_U S \\
 C_2 &\rightarrow SY_ ) \\
 Y_* &\rightarrow * \\
 Y_U &\rightarrow U \\
 Y_ ( &\rightarrow ( \\
 Y_ ) &\rightarrow )
 \end{aligned}$$

$V_{ii+7}$	$V_{18}$							
$V_{ii+6}$	$V_{17}$	$V_{28}$						
$V_{ii+5}$	$V_{16}$	$V_{27}$	$V_{38}$					
$V_{ii+4}$	$V_{15}$	$V_{26}$	$V_{37}$	$V_{48}$				
$V_{ii+3}$	$V_{14}$	$V_{25}$	$V_{36}$	$V_{47}$	$V_{58}$			
$V_{ii+2}$	$V_{13}$	$V_{24}$	$V_{35}$	$V_{46}$	$V_{57}$	$V_{68}$		
$V_{ii+1}$	$V_{12}$	$V_{23}$	$V_{34}$	$V_{45}$	$V_{56}$	$V_{67}$	$V_{78}$	
$V_{ii}$	$V_{11}$	$V_{22}$	$V_{33}$	$V_{44}$	$V_{55}$	$V_{66}$	$V_{77}$	$V_{88}$
$i$	1	2	3	4	5	6	7	8

# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$

$V_{ii+7}$	$V_{18}$							
$V_{ii+6}$	$V_{17}$	$V_{28}$						
$V_{ii+5}$	$V_{16}$	$V_{27}$	$V_{38}$					
$V_{ii+4}$	$V_{15}$	$V_{26}$	$V_{37}$	$V_{48}$				
$V_{ii+3}$	$V_{14}$	$V_{25}$	$V_{36}$	$V_{47}$	$V_{58}$			
$V_{ii+2}$	$V_{13}$	$V_{24}$	$V_{35}$	$V_{46}$	$V_{57}$	$V_{68}$		
$V_{ii+1}$	$V_{12}$	$V_{23}$	$V_{34}$	$V_{45}$	$V_{56}$	$V_{67}$	$V_{78}$	
$V_{ii}$	$S$	$Y_*$	$Y \cup$	$Y($	$S$	$S$	$Y)$	$Y_*$
	1	*	$\cup$	(	0	1	)	*
$i$	1	2	3	4	5	6	7	8

Fall  $\ell > 0$ :

- Jede Ableitung von  $w_i \dots w_j$  muss mit einer Regel der Form

$$A \rightarrow BC$$

beginnen, wobei ein  $k \in \{i, \dots, j - 1\}$  existiert mit

- $B \xrightarrow{*} w_i \dots w_k$  und
- $C \xrightarrow{*} w_{k+1} \dots w_j$ .

## Verfahren

- Speichere alle Mengen  $V_{rs}$  als Arrays der Länge  $|V|$ , in denen für jedes  $A \in V$  markiert ist, ob  $A \in V_{rs}$ .
- **Berechnung von  $V_{ij}$ :**  
Überprüfe für jede Regel  $(A \rightarrow BC) \in R$  und jedes  $k$ , ob

$$B \xrightarrow{*} w_i \dots w_k$$

$$C \xrightarrow{*} w_{k+1} \dots w_j$$

durch Ansehen der Stelle

- $B$  im Array zu  $V_{ik}$  und
- $C$  im Array zu  $V_{k+1 j}$ .

## Verfahren

- Speichere alle Mengen  $V_{rs}$  als Arrays der Länge  $|V|$ , in denen für jedes  $A \in V$  markiert ist, ob  $A \in V_{rs}$ .
- **Berechnung von  $V_{ij}$ :**  
Überprüfe für jede Regel  $(A \rightarrow BC) \in R$  und jedes  $k$ , ob

$$B \xrightarrow{*} w_j \dots w_k$$

$$C \xrightarrow{*} w_{k+1} \dots w_j$$

durch Ansehen der Stelle

- $B$  im Array zu  $V_{jk}$  und
- $C$  im Array zu  $V_{k+1 j}$ .

## Bemerkung

- Dies benötigt Aufwand in  $\mathcal{O}(n \cdot |R|)$



# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$

$V_{ii+7}$	$V_{18}$							
$V_{ii+6}$	$V_{17}$	$V_{28}$						
$V_{ii+5}$	$V_{16}$	$V_{27}$	$V_{38}$					
$V_{ii+4}$	$V_{15}$	$V_{26}$	$V_{37}$	$V_{48}$				
$V_{ii+3}$	$V_{14}$	$V_{25}$	$V_{36}$	$V_{47}$	$V_{58}$			
$V_{ii+2}$	$V_{13}$	$V_{24}$	$V_{35}$	$V_{46}$	$V_{57}$	$V_{68}$		
$V_{ii+1}$	$V_{12}$	$V_{23}$	$V_{34}$	$V_{45}$	$V_{56}$	$V_{67}$	$V_{78}$	
$V_{ii}$	$S$	$Y_*$	$Y \cup$	$Y($	$S$	$S$	$Y)$	$Y_*$
	1	*	$\cup$	(	0	1	)	*
$i$	1	2	3	4	5	6	7	8

# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$

$V_{ii+7}$	$V_{18}$							
$V_{ii+6}$	$V_{17}$	$V_{28}$						
$V_{ii+5}$	$V_{16}$	$V_{27}$	$V_{38}$					
$V_{ii+4}$	$V_{15}$	$V_{26}$	$V_{37}$	$V_{48}$				
$V_{ii+3}$	$V_{14}$	$V_{25}$	$V_{36}$	$V_{47}$	$V_{58}$			
$V_{ii+2}$	$V_{13}$	$V_{24}$	$V_{35}$	$V_{46}$	$V_{57}$	$V_{68}$		
$V_{ii+1}$	$S$	$\emptyset$	$\emptyset$	$\emptyset$	$S$	$C_2$	$\emptyset$	
$V_{ii}$	$S$	$Y_*$	$Y \cup$	$Y($	$S$	$S$	$Y)$	$Y_*$
	1	*	$\cup$	(	0	1	)	*
$i$	1	2	3	4	5	6	7	8

# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$

$V_{ii+7}$	$V_{18}$							
$V_{ii+6}$	$V_{17}$	$V_{28}$						
$V_{ii+5}$	$V_{16}$	$V_{27}$	$V_{38}$					
$V_{ii+4}$	$V_{15}$	$V_{26}$	$V_{37}$	$V_{48}$				
$V_{ii+3}$	$V_{14}$	$V_{25}$	$V_{36}$	$V_{47}$	$V_{58}$			
$V_{ii+2}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$C_2$	$\emptyset$		
$V_{ii+1}$	$S$	$\emptyset$	$\emptyset$	$\emptyset$	$S$	$C_2$	$\emptyset$	
$V_{ii}$	$S$	$Y_*$	$Y \cup$	$Y($	$S$	$S$	$Y)$	$Y_*$
	1	*	$\cup$	(	0	1	)	*
$i$	1	2	3	4	5	6	7	8

# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$

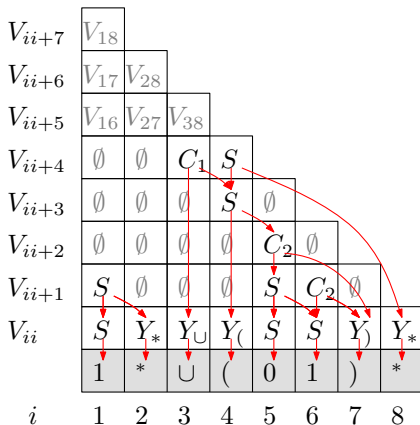
$V_{ii+7}$	$V_{18}$							
$V_{ii+6}$	$V_{17}$	$V_{28}$						
$V_{ii+5}$	$V_{16}$	$V_{27}$	$V_{38}$					
$V_{ii+4}$	$V_{15}$	$V_{26}$	$V_{37}$	$V_{48}$				
$V_{ii+3}$	$\emptyset$	$\emptyset$	$\emptyset$	$S$	$\emptyset$			
$V_{ii+2}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$C_2$	$\emptyset$		
$V_{ii+1}$	$S$	$\emptyset$	$\emptyset$	$\emptyset$	$S$	$C_2$	$\emptyset$	
$V_{ii}$	$S$	$Y_*$	$Y \cup$	$Y($	$S$	$S$	$Y)$	$Y_*$
	1	*	$\cup$	(	0	1	)	*
$i$	1	2	3	4	5	6	7	8

# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$

$V_{ii+7}$	$V_{18}$							
$V_{ii+6}$	$V_{17}$	$V_{28}$						
$V_{ii+5}$	$V_{16}$	$V_{27}$	$V_{38}$					
$V_{ii+4}$	$\emptyset$	$\emptyset$	$C_1$	$S$				
$V_{ii+3}$	$\emptyset$	$\emptyset$	$\emptyset$	$S$	$\emptyset$			
$V_{ii+2}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$C_2$	$\emptyset$		
$V_{ii+1}$	$S$	$\emptyset$	$\emptyset$	$\emptyset$	$S$	$C_2$	$\emptyset$	
$V_{ii}$	$S$	$Y_*$	$Y \cup$	$Y($	$S$	$S$	$Y)$	$Y_*$
	1	*	$\cup$	(	0	1	)	*
$i$	1	2	3	4	5	6	7	8

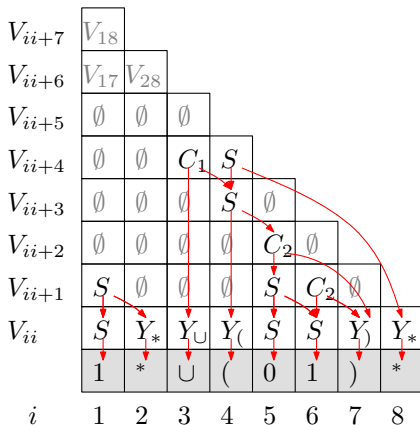


# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$

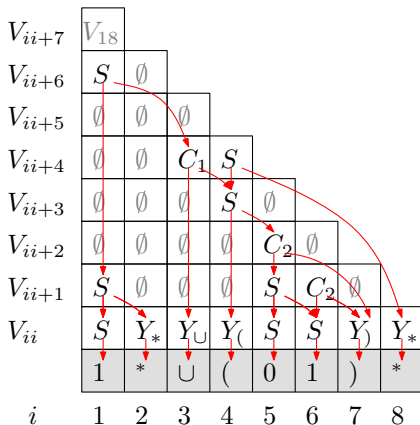
$V_{ii+7}$	$V_{18}$							
$V_{ii+6}$	$V_{17}$	$V_{28}$						
$V_{ii+5}$	$\emptyset$	$\emptyset$	$\emptyset$					
$V_{ii+4}$	$\emptyset$	$\emptyset$	$C_1$	$S$				
$V_{ii+3}$	$\emptyset$	$\emptyset$	$\emptyset$	$S$	$\emptyset$			
$V_{ii+2}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$C_2$	$\emptyset$		
$V_{ii+1}$	$S$	$\emptyset$	$\emptyset$	$\emptyset$	$S$	$C_2$	$\emptyset$	
$V_{ii}$	$S$	$Y_*$	$Y \cup$	$Y($	$S$	$S$	$Y)$	$Y_*$
	1	*	$\cup$	(	0	1	)	*
$i$	1	2	3	4	5	6	7	8



# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

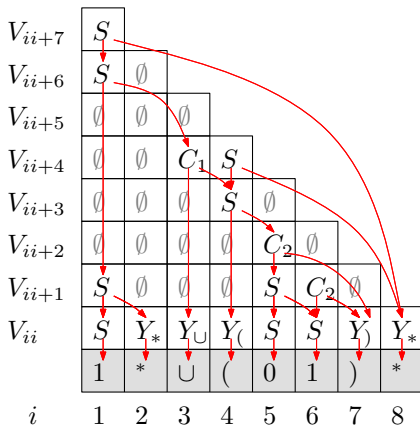
$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$



# CYK-Algorithmus – Beispiel

$$w = 1^* \cup (01)^*$$

$$\begin{aligned}
 S &\rightarrow SY_* \mid SC_1 \mid \\
 &\quad SS \mid Y(C_2 \mid \\
 &\quad e \mid \emptyset \mid 0 \mid 1 \\
 C_1 &\rightarrow Y \cup S \\
 C_2 &\rightarrow SY) \\
 Y_* &\rightarrow * \\
 Y \cup &\rightarrow \cup \\
 Y( &\rightarrow ( \\
 Y) &\rightarrow )
 \end{aligned}$$





# Ergebnisse zum Wortproblem

- **Typ-0 Grammatik.** Das Wortproblem ist nicht entscheidbar.
- **Typ-1 Grammatik.** Das Wortproblem ist NP-vollständig.
- **Typ-2 Grammatik.** Das Wortproblem ist in polynomieller Zeit lösbar.
- **Typ-3 Grammatik.** Das Wortproblem ist in linearer Zeit lösbar.