Theoretische Grundlagen der Informatik Übung

7. Übungstermin · 16. Januar 2018

Torsten Ueckerdt

Institut für Theoretische Informatik · Lehrstuhl Algorithmik

Übersicht

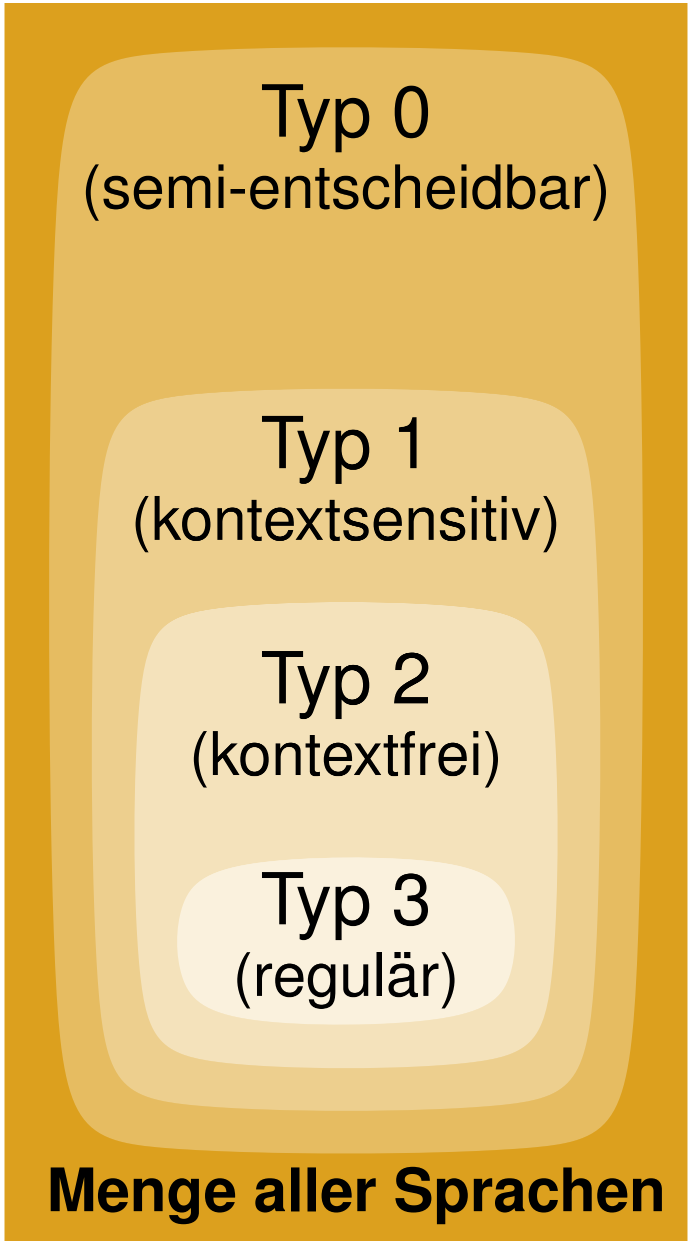
Inhalt

Chomsky Hierarchie und Grammatiken

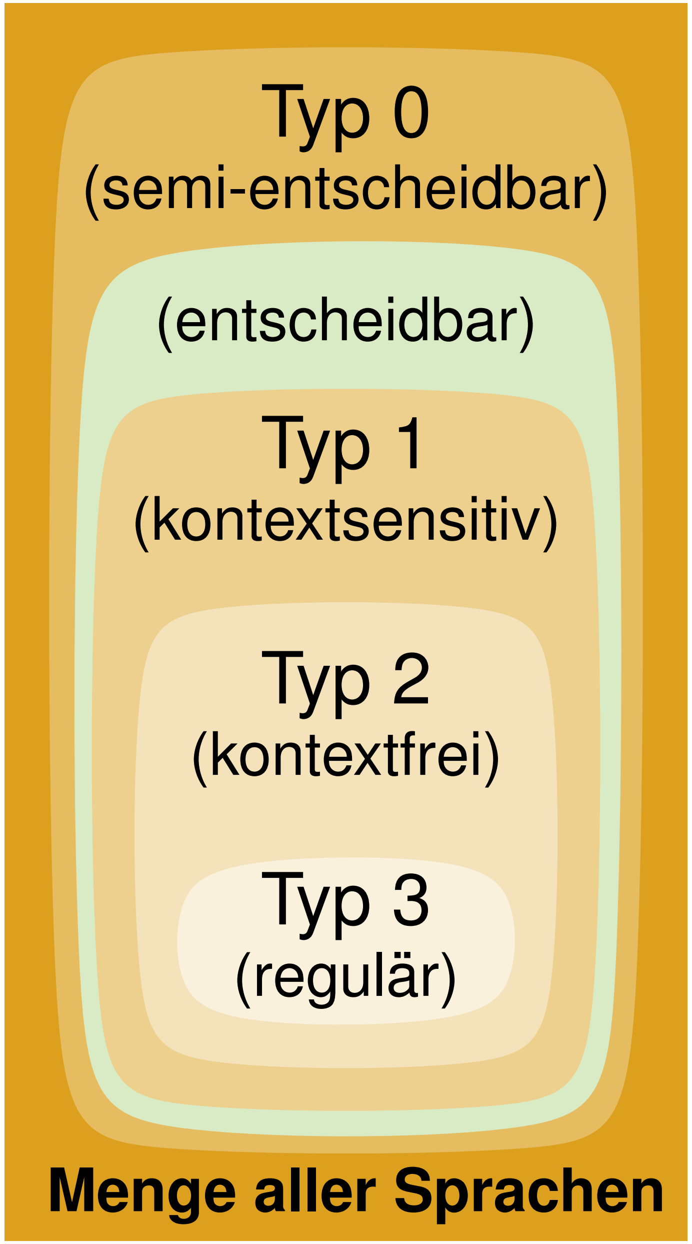
- Chomsky-0-Grammatiken und DTM's
- Konstruktion von Grammatiken
- Sprache der korrekten Klammerausdrücke
- Chomsky-Normalform
- Der Cocke-Younger-Kasami Algorithmus
- NEA aus Chomsky-3-Grammatik
- Eindeutige und mehrdeutige Grammatiken

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

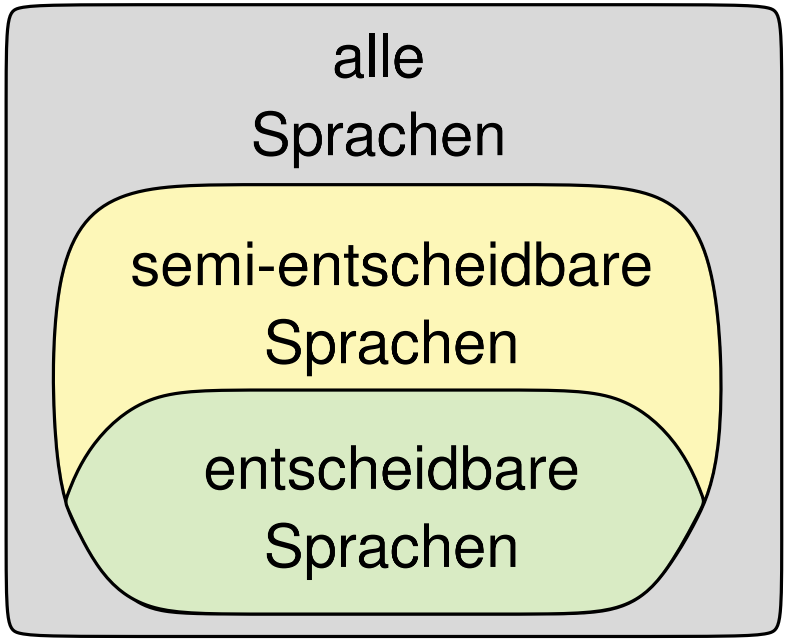
Chomsky Hierarchie



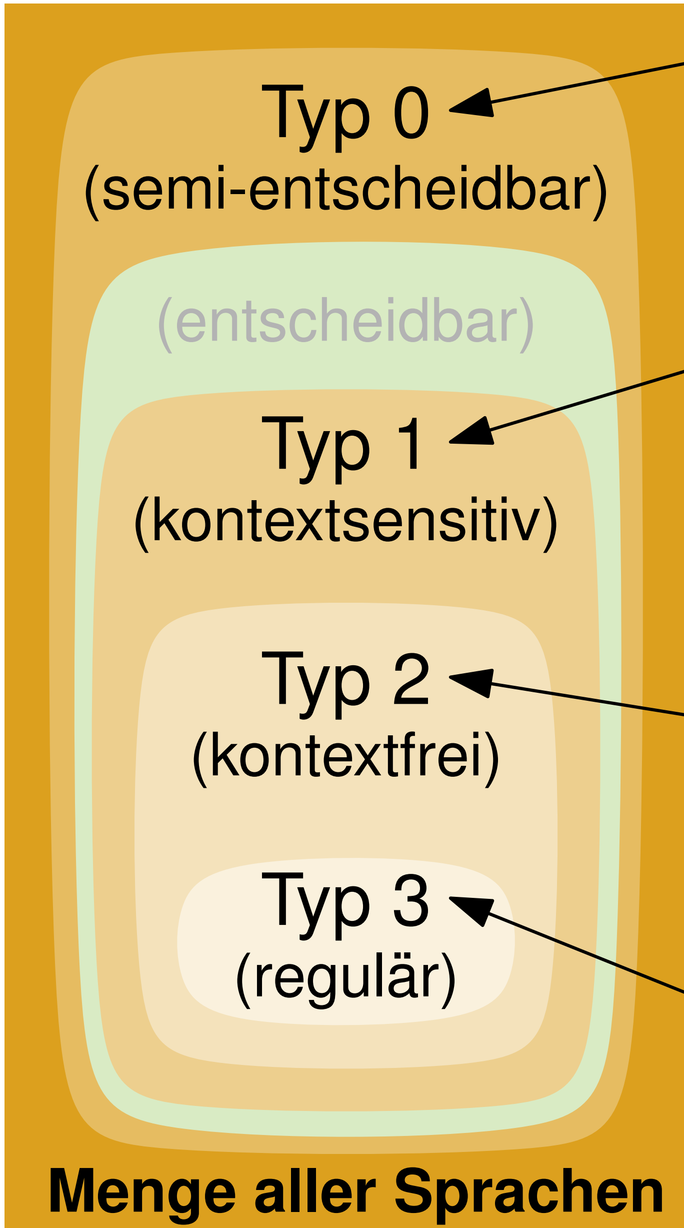
Chomsky Hierarchie



aus Übung 3:



Chomsky Hierarchie



Chomsky-0-Gram.

$G = (\Sigma, V, S, R)$

R beliebig

Chomsky-1-Gram.

$u \to w, |u| \leq |w|$

$u \in V^+, S \notin w$

$S \to \varepsilon$

Chomsky-2-Gram.

$A \to w, A \in V$

w beliebig

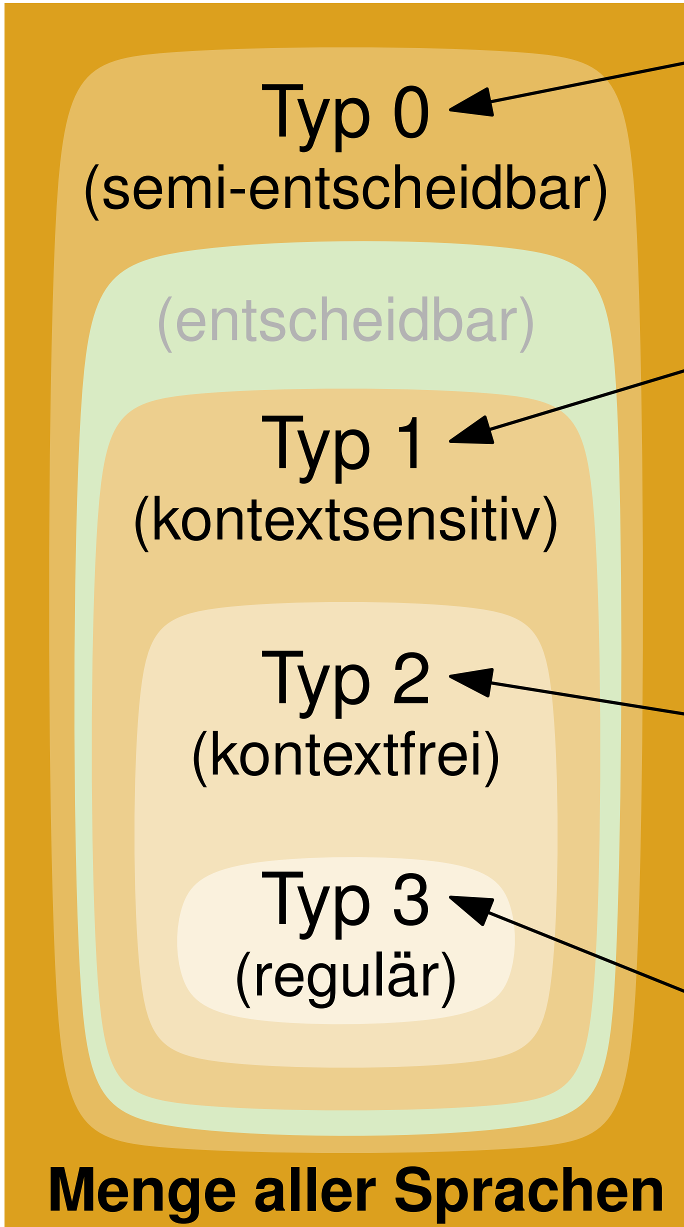
Chomsky-3-Gram.

$A \to w, A \in V$

$w \in \{\varepsilon\} \cup \Sigma \cdot V$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky Hierarchie



Chomsky-0-Gram.

$G = (\Sigma, V, S, R)$

R beliebig

NTM akzeptiert L

DTM akzeptiert L

Chomsky-1-Gram.

$u \rightarrow w, |u| \leq |w|$

$u \in V^{+}, S \notin w$

$S \rightarrow \varepsilon$

NTM mit Platzbedarf n erkennt Wörter d. Länge n in L

$\Rightarrow \mathcal{NTAPE}(n)$

Chomsky-2-Gram.

$A \rightarrow w, A \in V$

w beliebig

CYK-Alg. erkennt L in polynom. Zeit

$\Rightarrow$ Chomsky-Normalform

Chomsky-3-Gram.

$A \rightarrow w, A \in V$

$w \in \{\varepsilon\} \cup \Sigma \cdot V$

NEA erkennt L

DEA erkennt L

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky Hierarchie

Typ 0
(semi-entscheidbar)
(entscheidbar)
Typ 1
(kontextsensitiv)
Typ 2
(kontextfrei)
Typ 3
(regulär)
Menge aller Sprachen

Chomsky-0-Gram.
$G = (\Sigma, V, S, R)$
R beliebig

NTM akzeptiert L
DTM akzeptiert L

Chomsky-1-Gram.
$u \to w$, $|u| \leq |w|$
$u \in V^+$, $S \notin w$
$S \to \varepsilon$

NTM mit Platzbedarf n erkennt Wörter d. Länge n in L
$\Rightarrow \mathcal{NTAPE}(n)$

Chomsky-2-Gram.
$A \to w$, $A \in V$
w beliebig

CYK-Alg. erkennt L in polynom. Zeit
$\Rightarrow$ Chomsky-Normalform

Chomsky-3-Gram.
$A \to w$, $A \in V$
$w \in \{\varepsilon\} \cup \Sigma \cdot V$

NEA erkennt L
DEA erkennt L

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-0-Grammatiken und DTM's

Satz aus der Vorlesung (Vorlesung 13, 19.12.2017)**:** A $\Leftrightarrow$ B

Die von Typ-0-Grammatiken erzeugten Sprachen sind genau die von nichtdeterministischen Turingmaschinen akzeptierten Sprachen.

L ist Typ 0, d.h., es gibt Grammatik G A mit $L(G) = L$.	$\Leftrightarrow$	Es gibt nichtdeterm. TM $\mathcal{M}$ die L B akzeptiert.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-0-Grammatiken und DTM's

Satz aus der Vorlesung (Vorlesung 13, 19.12.2017)**:** A $\Leftrightarrow$ B

Die von Typ-0-Grammatiken erzeugten Sprachen sind genau die von nichtdeterministischen Turingmaschinen akzeptierten Sprachen.

L ist Typ 0, d.h., es gibt Grammatik G A mit $L(G) = L$. $\Leftrightarrow$ Es gibt nichtdeterm. B TM $\mathcal{M}$ die L akzeptiert.

Beweis aus der Vorlesung von A $\Rightarrow$ B **:** Wir konstruieren eine NTM die L akzeptiert.

- Schreibe S auf das Band.
- **Repeat**
  - Wähle nichtdeterministisch eine anwendbare Ableitungsregel.
  - Vergleiche das erzeugte Wort mit der Eingabe w.
  - Bei Gleichheit wird w akzeptiert.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-0-Grammatiken und DTM's

Satz aus der Vorlesung (Vorlesung 13, 19.12.2017): A $\Leftrightarrow$ B

Die von Typ-0-Grammatiken erzeugten Sprachen sind genau die von nichtdeterministischen Turingmaschinen akzeptierten Sprachen.

L ist Typ 0, d.h., es gibt Grammatik G (A) mit $L(G) = L$.	$\Leftrightarrow$	Es gibt nichtdeterm. TM $\mathcal{M}$ die L (B) akzeptiert.	$\Leftrightarrow$	(C) Es gibt deterministische TM $\mathcal{M}$ die L akzeptiert.

Beweis aus der Vorlesung von A $\Rightarrow$ B.

In der Vorlesung wurde dann C $\Rightarrow$ A **bewiesen.**

Und es gilt auch allgemein, dass B $\Rightarrow$ C.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-0-Grammatiken und DTM's

Satz aus der Vorlesung (Vorlesung 13, 19.12.2017): A $\Leftrightarrow$ B

Die von Typ-0-Grammatiken erzeugten Sprachen sind genau die von nichtdeterministischen Turingmaschinen akzeptierten Sprachen.

L ist Typ 0, d.h., es gibt Grammatik G (A) mit $L(G) = L$.	$\Leftrightarrow$	Es gibt nichtdeterm. TM $\mathcal{M}$ die L (B) akzeptiert.	$\Leftrightarrow$	(C) Es gibt deterministische TM $\mathcal{M}$ die L akzeptiert.

Beweis aus der Vorlesung von A $\Rightarrow$ B.

In der Vorlesung wurde dann C $\Rightarrow$ A **bewiesen.**

Und es gilt auch allgemein, dass B $\Rightarrow$ C .

DTM's und NTM's sind gleichmächtig. (Wenn Laufzeit egal ist.)

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-0-Grammatiken und DTM's

Satz aus der Vorlesung (Vorlesung 13, 19.12.2017)**:** A $\Leftrightarrow$ B
Die von Typ-0-Grammatiken erzeugten Sprachen sind genau die von nichtdeterministischen Turingmaschinen akzeptierten Sprachen.

L ist Typ 0, d.h., es gibt Grammatik G A mit $L(G) = L$. $\Leftrightarrow$ Es gibt nichtdeterm. TM $\mathcal{M}$ die L B akzeptiert. $\Leftrightarrow$ C Es gibt deterministische TM $\mathcal{M}$ die L akzeptiert.

Beweis aus der Vorlesung von A $\Rightarrow$ B **:** Wir konstruieren eine NTM die L akzeptiert.

- Schreibe S auf das Band.
- **Repeat**
  - Wähle nichtdeterministisch eine anwendbare Ableitungsregel.
  - Vergleiche das erzeugte Wort mit der Eingabe w.
  - Bei Gleichheit wird w akzeptiert.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-0-Grammatiken und DTM's

Satz aus der Vorlesung (Vorlesung 13, 19.12.2017): A $\Leftrightarrow$ B
Die von Typ-0-Grammatiken erzeugten Sprachen sind genau die von nichtdeterministischen Turingmaschinen akzeptierten Sprachen.

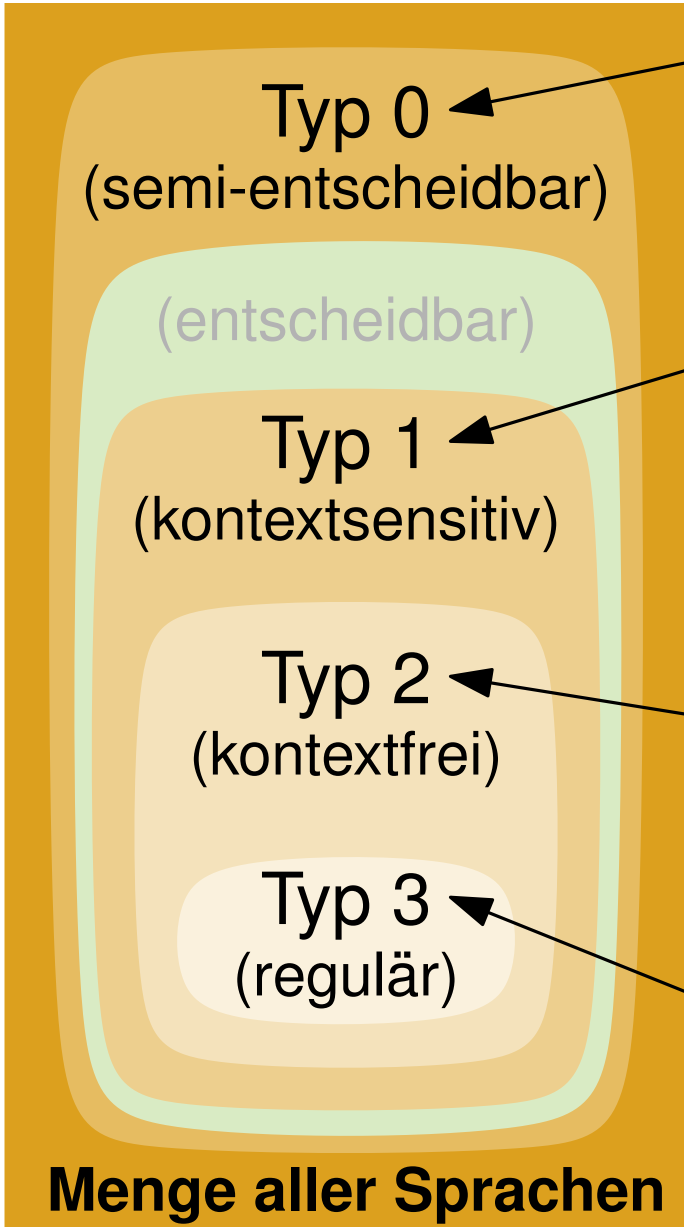
A L ist Typ 0, d.h., es gibt Grammatik G mit $L(G) = L$. $\Leftrightarrow$ B Es gibt nichtdeterm. TM $\mathcal{M}$ die L akzeptiert. $\Leftrightarrow$ C Es gibt deterministische TM $\mathcal{M}$ die L akzeptiert.

Hier beweisen wir A $\Rightarrow$ C **:** Wir konstruieren eine DTM die L akzeptiert.

- **For** $i = 1, 2, 3, \ldots$
  - Schreibe S auf das Band.
  - Wähle deterministisch eine Folge von genau i Ableitungsregeln.
  - Wenn anwendbar, vergleiche erzeugtes Wort mit der Eingabe w.
  - Bei Gleichheit wird w akzeptiert, ansonsten das Band gelöscht.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky Hierarchie



Chomsky-0-Gram.

$G = (\Sigma, V, S, R)$

R beliebig

NTM akzeptiert L

DTM akzeptiert L

Chomsky-1-Gram.

$u \to w, |u| \leq |w|$

$u \in V^+, S \notin w$

$S \to \varepsilon$

NTM mit Platzbedarf n erkennt Wörter d. Länge n in L

$\Rightarrow \mathcal{NTAPE}(n)$

Chomsky-2-Gram.

$A \to w, A \in V$

w beliebig

CYK-Alg. erkennt L in polynom. Zeit

$\Rightarrow$ Chomsky-Normalform

Chomsky-3-Gram.

$A \to w, A \in V$

$w \in \{\varepsilon\} \cup \Sigma \cdot V$

NEA erkennt L

DEA erkennt L

Chomsky Hierarchie

Typ 0
(semi-entscheidbar)

(entscheidbar)

Typ 1
(kontextsensitiv)

Typ 2
(kontextfrei)

Typ 3
(regulär)

Menge aller Sprachen

Chomsky-0-Gram.

$G = (\Sigma, V, S, R)$

R beliebig

NTM akzeptiert L

DTM akzeptiert L

Chomsky-1-Gram.

$u \rightarrow w, |u| \leq |w|$

$u \in V^+, S \notin w$

$S \rightarrow \varepsilon$

NTM mit Platzbedarf n erkennt Wörter d. Länge n in L

$\Rightarrow \mathcal{NTAPE}(n)$

Chomsky-2-Gram.

$A \rightarrow w, A \in V$

w beliebig

CYK-Alg. erkennt L in polynom. Zeit

$\Rightarrow$ Chomsky-Normalform

Chomsky-3-Gram.

$A \rightarrow w, A \in V$

$w \in \{\varepsilon\} \cup \Sigma \cdot V$

NEA erkennt L

DEA erkennt L

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky Hierarchie

Menge aller Sprachen

Typ 0 (semi-entscheidbar)

(entscheidbar)

Typ 1 (kontextsensitiv)

Typ 2 (kontextfrei)

Typ 3 (regulär)

Chomsky-0-Gram.

$G = (\Sigma, V, S, R)$

R beliebig

NTM akzeptiert L
DTM akzeptiert L

Chomsky-1-Gram.

$u \to w, |u| \leq |w|$

$u \in V^+, S \notin w$

$S \to \varepsilon$

NTM mit Platzbedarf n erkennt Wörter d. Länge n in L

$\Rightarrow \mathcal{NTAPE}(n)$

Chomsky-2-Gram.

$A \to w, A \in V$

w beliebig

CYK-Alg. erkennt L in polynom. Zeit

$\Rightarrow$ Chomsky-Normalform

Chomsky-3-Gram.

$A \to w, A \in V$

$w \in \{\varepsilon\} \cup \Sigma \cdot V$

NEA erkennt L
DEA erkennt L

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

$$L_2 = \{a^i b^j c^k \in \{a, b, c\}^\star \mid i = j \vee j = k\}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

$$L_2 = \{a^i b^j c^k \in \{a, b, c\}^\star \mid i = j \vee j = k\}$$

1. Schritt: Von welchem Typ ist die Sprache?

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

$$L_2 = \{a^i b^j c^k \in \{a, b, c\}^\star \mid i = j \vee j = k\}$$

1. Schritt: Von welchem Typ ist die Sprache? Typ-2

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

$$L_2 = \{a^i b^j c^k \in \{a, b, c\}^\star \mid i = j \vee j = k\}$$

1. Schritt: Von welchem Typ ist die Sprache? Typ-2

2. Schritt: Grammatik vom Typ-2 formal definieren und formulieren.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

$$L_2 = \{a^i b^j c^k \in \{a, b, c\}^\star \mid i = j \vee j = k\}$$

1. Schritt: Von welchem Typ ist die Sprache? Typ-2

2. Schritt: Grammatik vom Typ-2 formal definieren und formulieren.

$$G = (\Sigma, V, S, R) \qquad \Sigma = \{a, b, c\} \qquad V = \{S, T, U, V, W\}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

$$L_2 = \{a^i b^j c^k \in \{a, b, c\}^\star \mid i = j \vee j = k\}$$

1. Schritt: Von welchem Typ ist die Sprache? Typ-2

2. Schritt: Grammatik vom Typ-2 formal definieren und formulieren.

$$G = (\Sigma, V, S, R) \qquad \Sigma = \{a, b, c\} \qquad V = \{S, T, U, V, W\}$$

$$R = \{ \begin{aligned} S &\rightarrow TU \mid VW, \\ T &\rightarrow aTb \mid \varepsilon, \\ U &\rightarrow Uc \mid \varepsilon, \\ W &\rightarrow bWc \mid \varepsilon, \\ V &\rightarrow Va \mid \varepsilon. \end{aligned} \}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

a) $L_1 = \{ww \mid w \in \{0, 1\}^\star\}$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgende Sprache eine Grammatik an.

a) $L_1 = \{ww \mid w \in \{0, 1\}^\star\}$

1. Schritt: Von welchem Typ ist die Sprache?

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

a) $L_1 = \{ww \mid w \in \{0,1\}^\star\}$

1. Schritt: Von welchem Typ ist die Sprache? Typ-1

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

a) $L_1 = \{ww \mid w \in \{0,1\}^\star\}$

1. **Schritt:** Von welchem Typ ist die Sprache? Typ-1

2. **Schritt:** Grammatik vom Typ-1 formal definieren und formulieren.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

a) $L_1 = \{ww \mid w \in \{0,1\}^\star\}$

1. Schritt: Von welchem Typ ist die Sprache? Typ-1

2. Schritt: Grammatik vom Typ-1 formal definieren und formulieren.

$$G = (\Sigma, V, S, R) \qquad \Sigma = \{0,1\} \quad V = \{S, X_i, R_i, L_i, T\}$$

$$\text{für } i \in \{0,1\}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Konstruktion von Grammatiken

Geben Sie für die folgendene Sprache eine Grammatik an.

a) $L_1 = \{ww \mid w \in \{0,1\}^\star\}$

1. Schritt: Von welchem Typ ist die Sprache? Typ-1

2. Schritt: Grammatik vom Typ-1 formal definieren und formulieren.

$G = (\Sigma, V, S, R)$ $\Sigma = \{0,1\}$ $V = \{S, X_i, R_i, L_i, T\}$ für $i \in \{0,1\}$

$$R = \{ \; S \rightarrow T \mid \varepsilon$$

$$T \rightarrow 0TX_0 \mid 1TX_1 \mid L_0R_0 \mid L_1R_1$$

$$\begin{array}{lll} R_0X_0 \rightarrow X_0R_0 & L_0X_0 \rightarrow L_0R_0 & R_0 \rightarrow 0 \\ R_0X_1 \rightarrow X_1R_0 & L_0X_1 \rightarrow L_0R_1 & R_1 \rightarrow 1 \\ R_1X_0 \rightarrow X_0R_1 & L_1X_0 \rightarrow L_1R_0 & L_0 \rightarrow 0 \\ R_1X_1 \rightarrow X_1R_1 & L_1X_1 \rightarrow L_1R_1 & L_1 \rightarrow 1 \end{array}$$

$$\}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(a) Konstruieren Sie eine Typ-k-Grammatik mit maximalem k die $L_{()}$ erzeugt.

(b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

(c) Beweisen Sie die Maximalität von k.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(a) Konstruieren Sie eine Typ-k-Grammatik mit maximalem k die $L_{()}$ erzeugt.

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{ (\,,\,) \}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(a) Konstruieren Sie eine Typ-k-Grammatik mit maximalem k die $L_{()}$ erzeugt.

1. Schritt: Von welchem Typ ist die Grammatik?

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(a) Konstruieren Sie eine Typ-k-Grammatik mit maximalem k die $L_{()}$ erzeugt.

1. Schritt: Von welchem Typ ist die Grammatik? Typ-2 ✓

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(a) Konstruieren Sie eine Typ-k-Grammatik mit maximalem k die $L_{()}$ erzeugt.

1. Schritt: Von welchem Typ ist die Grammatik? Typ-2 ✓

2. Schritt: Grammatik vom Typ-2 formal definieren und formulieren.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

a) Konstruieren Sie eine Typ-k-Grammatik mit maximalem k die $L_{()}$ erzeugt.

1. Schritt: Von welchem Typ ist die Grammatik? Typ-2 ✓

2. Schritt: Grammatik vom Typ-2 formal definieren und formulieren.

$$G = (\Sigma, V, S, R) \qquad \Sigma = \{(,)\} \qquad V = \{S\}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

a) Konstruieren Sie eine Typ-k-Grammatik mit maximalem k die $L_{()}$ erzeugt.

1. Schritt: Von welchem Typ ist die Grammatik? Typ-2 ✓

2. Schritt: Grammatik vom Typ-2 formal definieren und formulieren.

$$G = (\Sigma, V, S, R) \qquad \Sigma = \{(,)\} \qquad V = \{S\}$$

$$R = \{$$

$$S \to (S)S \mid \varepsilon$$

$$\}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{ (,) \}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

$L(G) \subseteq L_{()}$**:**

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

$L(G) \subseteq L_{()}$**:**

- Klammern werden nie gelöscht oder verschoben (Typ-2)

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammer-ausdrücke gegeben.

(b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

$L(G) \subseteq L_{()}$:

- Klammern werden nie gelöscht oder verschoben (Typ-2)
- Jede Produktion, die eine Klammer erzeugt, erzeugt genau ein Paar (und). Also hat jedes Wort in $L(G)$ genauso viele öffnende wie schließende Klammern.

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

$L(G) \subseteq L_{()}$**:**

- Klammern werden nie gelöscht oder verschoben (Typ-2)
- Jede Produktion, die eine Klammer erzeugt, erzeugt genau ein Paar (und). Also hat jedes Wort in $L(G)$ genauso viele öffnende wie schließende Klammern.
- Jede Produktion, die mindestens eine Klammer erzeugt, erzeugt (vor). Also beinhaltet für jedes Wort w in $L(G)$ jedes Präfix von w mindestens so viele öffnende wie schließende Klammern.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{ (\,,\,) \}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

$L(G) \supseteq L_{()}$**:**

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{\,(\,,\,)\,\}$ ist die Sprache $L_{()}$ der korrekten Klammer-ausdrücke gegeben.

b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

$L(G) \supseteq L_{()}$**:**

- Sei w in $L_{()}$ beliebig.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{ (\,,\,) \}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

$L(G) \supseteq L_{()}$**:**

- Sei w in $L_{()}$ beliebig.
- Algorithmus, der w erzeugt:

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammer-ausdrücke gegeben.

b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

$L(G) \supseteq L_{()}$:

- Sei w in $L_{()}$ beliebig.
- Algorithmus, der w erzeugt:
  - $w = (w_1)w_2$, (w_1) ist erstes Klammerpaar im Ausdruck
    - Führe $S \to (S)S$ aus

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

b) Zeigen Sie, dass Ihre Grammatik genau $L_{()}$ erzeugt.

Zu zeigen ist: $L(G) = L_{()}$

$L(G) \supseteq L_{()}$:

- Sei w in $L_{()}$ beliebig.
- Algorithmus, der w erzeugt:
  - $w = (w_1)w_2$, (w_1) ist erstes Klammerpaar im Ausdruck
    - Führe $S \to (S)S$ aus
  - $w = \varepsilon$
    - Führe $S \to \varepsilon$ aus

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(c) Beweisen Sie die Maximalität von k.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(c) Beweisen Sie die Maximalität von k.

- $L_{()}$ ist nicht regulär (Pumping Lemma)

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(c) Beweisen Sie die Maximalität von k.

- $L_{()}$ ist nicht regulär (Pumping Lemma)
- reguläre Sprachen entsprechen Chomsky-Typ-3

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Sprache der korrekten Klammerausdrücke

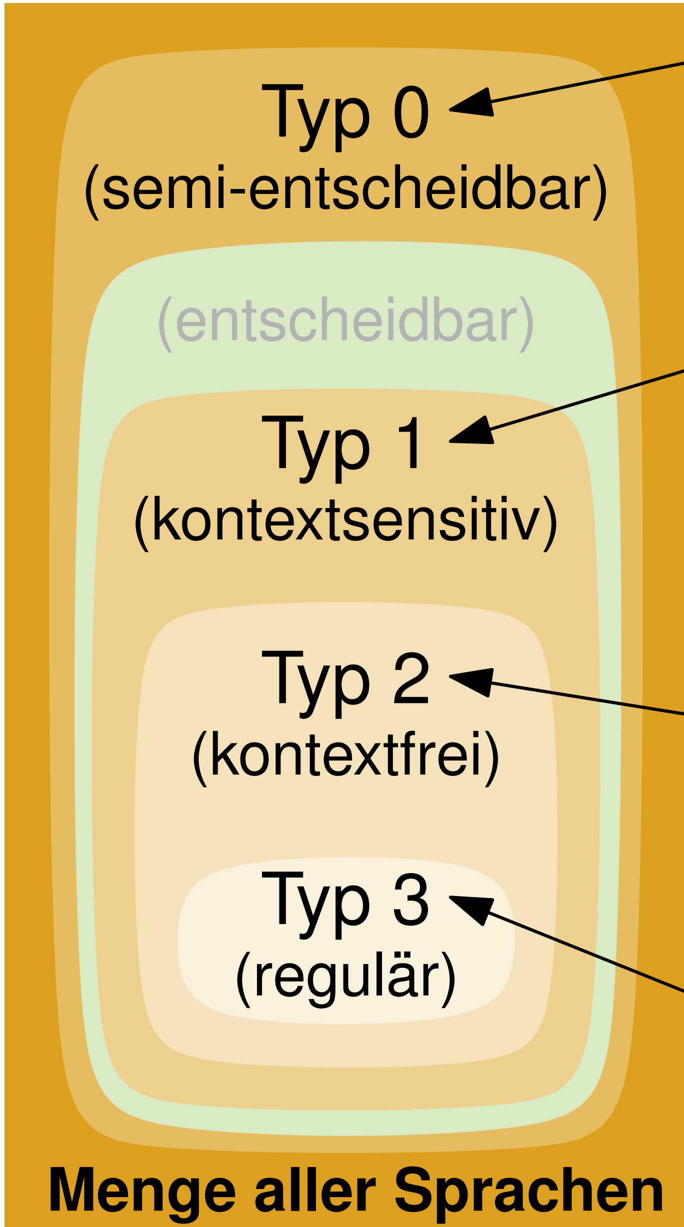
Über dem Alphabet $\Sigma = \{(,)\}$ ist die Sprache $L_{()}$ der korrekten Klammerausdrücke gegeben.

(c) Beweisen Sie die Maximalität von k.

- $L_{()}$ ist nicht regulär (Pumping Lemma)
- reguläre Sprachen entsprechen Chomsky-Typ-3
- $\Rightarrow$ Chomsky-Typ-2 ist maximal.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky Hierarchie



Chomsky-0-Gram.

$G = (\Sigma, V, S, R)$

R beliebig

NTM akzeptiert L

DTM akzeptiert L

Chomsky-1-Gram.

$u \to w, |u| \leq |w|$

$u \in V^+, S \notin w$

$S \to \varepsilon$

NTM mit Platzbedarf n erkennt Wörter d. Länge n in L

$\Rightarrow \mathcal{NTAPE}(n)$

Chomsky-2-Gram.

$A \to w, A \in V$

w beliebig

CYK-Alg. erkennt L in polynom. Zeit

$\Rightarrow$ Chomsky-Normalform

Chomsky-3-Gram.

$A \to w, A \in V$

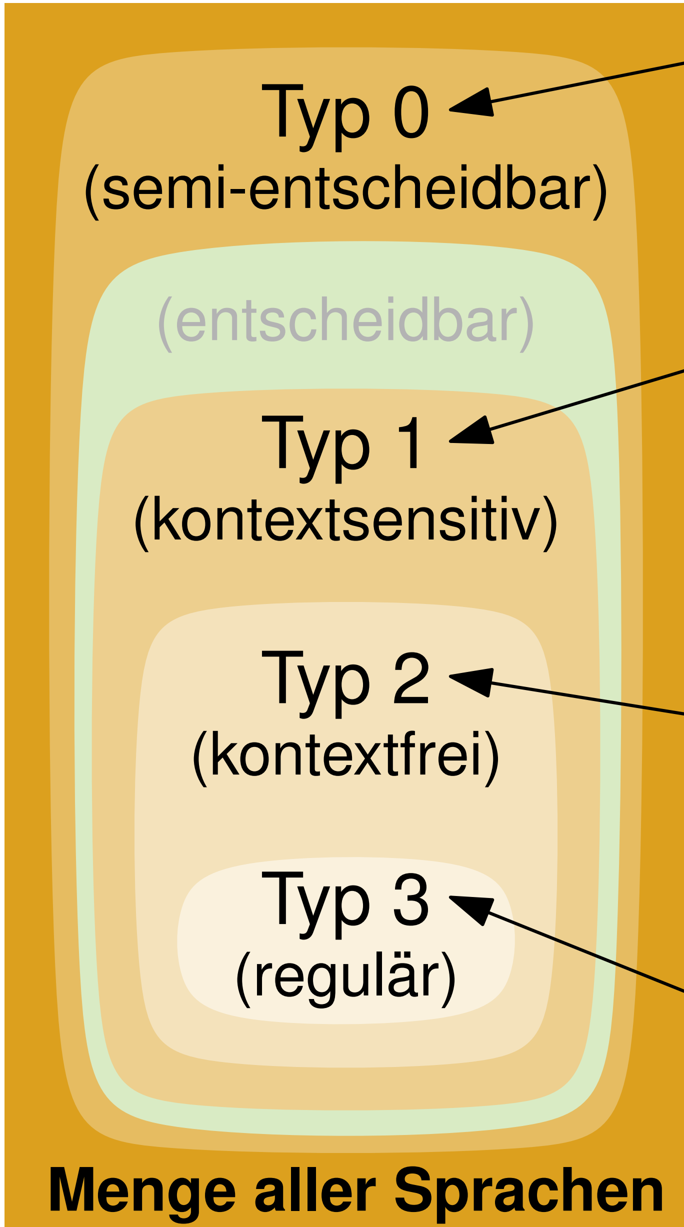
$w \in \{\varepsilon\} \cup \Sigma \cdot V$

NEA erkennt L

DEA erkennt L

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky Hierarchie



Chomsky-0-Gram.
$G = (\Sigma, V, S, R)$
R beliebig

NTM akzeptiert L
DTM akzeptiert L

Chomsky-1-Gram.
$u \to w, |u| \leq |w|$
$u \in V^+, S \notin w$
$S \to \varepsilon$

NTM mit Platzbedarf n erkennt Wörter d. Länge n in L
$\Rightarrow \mathcal{NTAPE}(n)$

Chomsky-2-Gram.
$A \to w, A \in V$
w beliebig

CYK-Alg. erkennt L in polynom. Zeit
$\Rightarrow$ Chomsky-Normalform

Chomsky-3-Gram.
$A \to w, A \in V$
$w \in \{\varepsilon\} \cup \Sigma \cdot V$

NEA erkennt L
DEA erkennt L

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

Gegeben sei die kontextfreie Grammatik $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \rightarrow A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \rightarrow S \mid B\mathtt{a}, \quad D \rightarrow \mathtt{d} \mid \mathtt{d}DD,$$
$$A \rightarrow B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \rightarrow D \mid \mathtt{c}.$$

a) Lässt sich der CYK-Algorithmus auf G (ohne Abänderungen) anwenden? Begründen Sie Ihre Antwort. Ändern Sie gegebenfalls G ab.

b) Prüfen Sie, ob das Wort $\mathtt{addd}$ in $L(G)$ liegt. Verwenden Sie dafür den CYK-Algorithmus.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

Gegeben sei die kontextfreie Grammatik $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

a) Lässt sich der CYK-Algorithmus auf G (ohne Abänderungen) anwenden? Begründen Sie Ihre Antwort. Ändern Sie gegebenfalls G ab.

b) Prüfen Sie, ob das Wort $\mathtt{addd}$ in $L(G)$ liegt. Verwenden Sie dafür den CYK-Algorithmus.

(a) **Nein. Begründung:**

- G ist *nicht* in Chomsky-Normalform

Chomsky-Normalform:

$A \to BC$ oder $A \to a$ mit $A, B, C \in V$ und $a \in \Sigma$

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \rightarrow A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \rightarrow S \mid B\mathtt{a}, \quad D \rightarrow \mathtt{d} \mid \mathtt{d}DD,$$
$$A \rightarrow B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \rightarrow D \mid \mathtt{c}.$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

1. Schritt: alle Regeln sind der Form

- $X \to Y, Y \in V^\star$ oder $X \to a, a \in \Sigma$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

1. Schritt: alle Regeln sind der Form

- $X \to Y, Y \in V^\star$ oder $X \to a, a \in \Sigma$

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C$$

$$B \to S \mid B\mathtt{a}$$

$$D \to \mathtt{d} \mid \mathtt{d}DD$$

$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}$$

$$C \to D \mid \mathtt{c}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

1. Schritt: alle Regeln sind der Form

- $X \to Y, Y \in V^\star$ oder $X \to a, a \in \Sigma$

$S \to A \mid \mathtt{a}B \mid \mathtt{a}C$	$S \to A \mid Z_aB \mid Z_aC$
$B \to S \mid B\mathtt{a}$	$B \to S \mid BZ_a$
$D \to \mathtt{d} \mid \mathtt{d}DD$	$D \to Z_d \mid Z_dDD$
$A \to B \mid C \mid \mathtt{c}A\mathtt{d}$	$A \to B \mid C \mid Z_cAZ_d$
$C \to D \mid \mathtt{c}$	$C \to D \mid Z_c$

$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

1. Schritt: alle Regeln sind der Form

- $X \to Y, Y \in V^\star$ oder $X \to a, a \in \Sigma$

$S \to A \mid \mathtt{a}B \mid \mathtt{a}C$	$S \to A \mid Z_a B \mid Z_a C$
$B \to S \mid B\mathtt{a}$	$B \to S \mid BZ_a$
$D \to \mathtt{d} \mid \mathtt{d}DD$	$D \to Z_d \mid Z_d DD$
$A \to B \mid C \mid \mathtt{c}A\mathtt{d}$	$A \to B \mid C \mid Z_c A Z_d$
$C \to D \mid \mathtt{c}$	$C \to D \mid Z_c$

$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

1. Schritt: alle Regeln sind der Form

- $X \to Y, Y \in V^\star$ oder $X \to a, a \in \Sigma$

$S \to A \mid \mathtt{a}B \mid \mathtt{a}C$	$S \to A \mid Z_a B \mid Z_a C$
$B \to S \mid B\mathtt{a}$	$B \to S \mid B Z_a$
$D \to \mathtt{d} \mid \mathtt{d}DD$	$D \to Z_d \mid Z_d DD$
$A \to B \mid C \mid \mathtt{c}A\mathtt{d}$	$A \to B \mid C \mid Z_c A Z_d$
$C \to D \mid \mathtt{c}$	$C \to D \mid Z_c$

$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

1. Schritt: alle Regeln sind der Form

- $X \to Y, Y \in V^\star$ oder $X \to a, a \in \Sigma$

$S \to A \mid \mathtt{a}B \mid \mathtt{a}C$	$S \to A \mid Z_a B \mid Z_a C$
$B \to S \mid B\mathtt{a}$	$B \to S \mid B Z_a$
$D \to \mathtt{d} \mid \mathtt{d}DD$	$D \to Z_d \mid Z_d DD$
$A \to B \mid C \mid \mathtt{c}A\mathtt{d}$	$A \to B \mid C \mid Z_c A Z_d$
$C \to D \mid \mathtt{c}$	$C \to D \mid Z_c$

$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

2. Schritt: Rechten Seiten haben Länge ≤ 2.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

2. Schritt: Rechten Seiten haben Länge ≤ 2.

$$S \to A \mid Z_a B \mid Z_a C$$

$$B \to S \mid B Z_a$$

$$D \to Z_d \mid Z_d DD$$

$$A \to B \mid C \mid Z_c A Z_d$$

$$C \to D \mid Z_c$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

2. Schritt: Rechten Seiten haben Länge ≤ 2.

$S \to A \mid Z_a B \mid Z_a C$	$S \to A \mid Z_a B \mid Z_a C$
$B \to S \mid B Z_a$	$B \to S \mid B Z_a$
$D \to Z_d \mid Z_d DD$	$D \to Z_d \mid Z_d E$
$A \to B \mid C \mid Z_c A Z_d$	$A \to B \mid C \mid Z_c F$
$C \to D \mid Z_c$	$C \to D \mid Z_c$
$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$	$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$
	$E \to DD \quad F \to A Z_d$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

2. Schritt: Rechten Seiten haben Länge ≤ 2.

$S \to A \mid Z_aB \mid Z_aC$	$S \to A \mid Z_aB \mid Z_aC$
$B \to S \mid BZ_a$	$B \to S \mid BZ_a$
$D \to Z_d \mid Z_dDD$	$D \to Z_d \mid Z_dE$
$A \to B \mid C \mid Z_cAZ_d$	$A \to B \mid C \mid Z_cF$
$C \to D \mid Z_c$	$C \to D \mid Z_c$
$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$	$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$
	$E \to DD \quad F \to AZ_d$

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

2. Schritt: Rechten Seiten haben Länge $\leq$ 2.

$S \to A \mid Z_aB \mid Z_aC$	$S \to A \mid Z_aB \mid Z_aC$
$B \to S \mid BZ_a$	$B \to S \mid BZ_a$
$D \to Z_d \mid Z_dDD$	$D \to Z_d \mid Z_dE$
$A \to B \mid C \mid Z_cAZ_d$	$A \to B \mid C \mid Z_cF$
$C \to D \mid Z_c$	$C \to D \mid Z_c$
$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$	$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$
	$E \to DD \quad F \to AZ_d$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

3. Schritt: Es kommen keine Regeln $A \to \varepsilon$ vor.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

3. Schritt: Es kommen keine Regeln $A \to \varepsilon$ vor.

$$S \to A \mid Z_a B \mid Z_a C$$

$$B \to S \mid B Z_a$$

$$D \to Z_d \mid Z_d E$$

$$A \to B \mid C \mid Z_c F$$

$$C \to D \mid Z_c$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

$$E \to DD \quad F \to A Z_d$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

3. Schritt: Es kommen keine Regeln $A \to \varepsilon$ vor.

$$S \to A \mid Z_aB \mid Z_aC$$

$$B \to S \mid BZ_a$$

$$D \to Z_d \mid Z_dE$$

$$A \to B \mid C \mid Z_cF$$

$$C \to D \mid Z_c$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

$$E \to DD \quad F \to AZ_d$$

✓

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

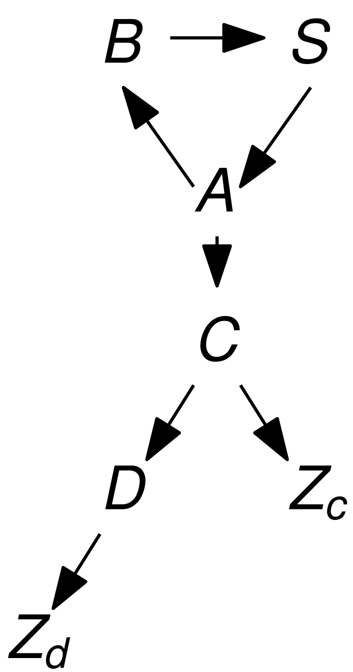
4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.



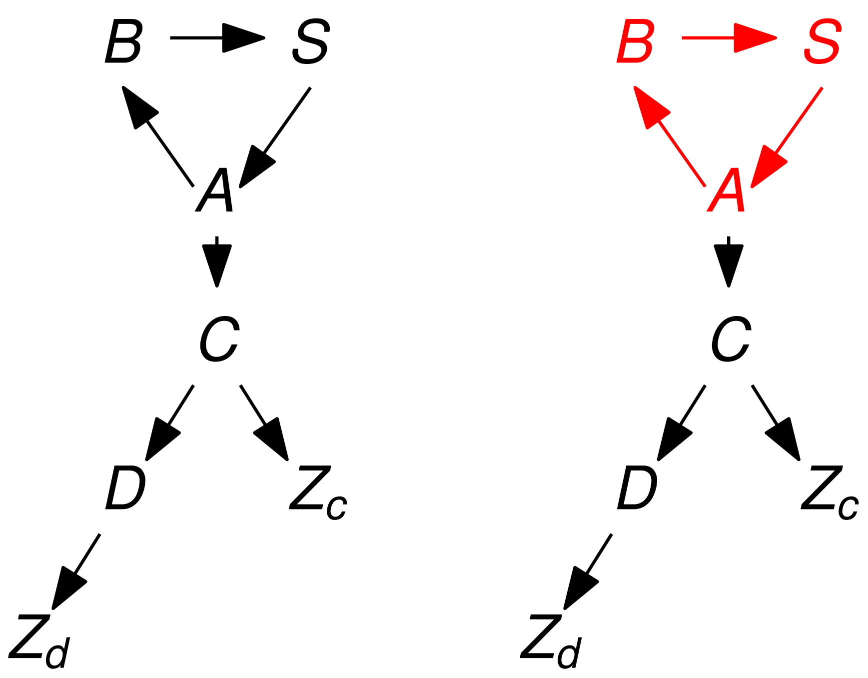
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.



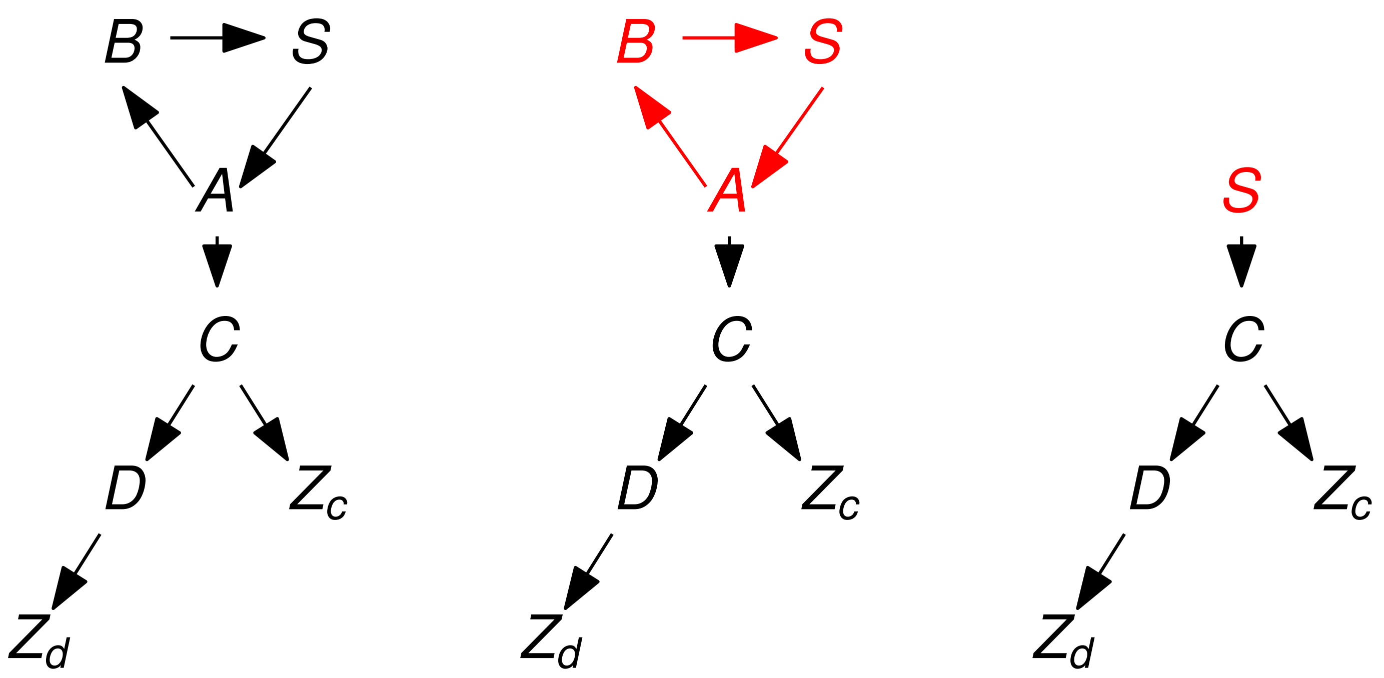
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.



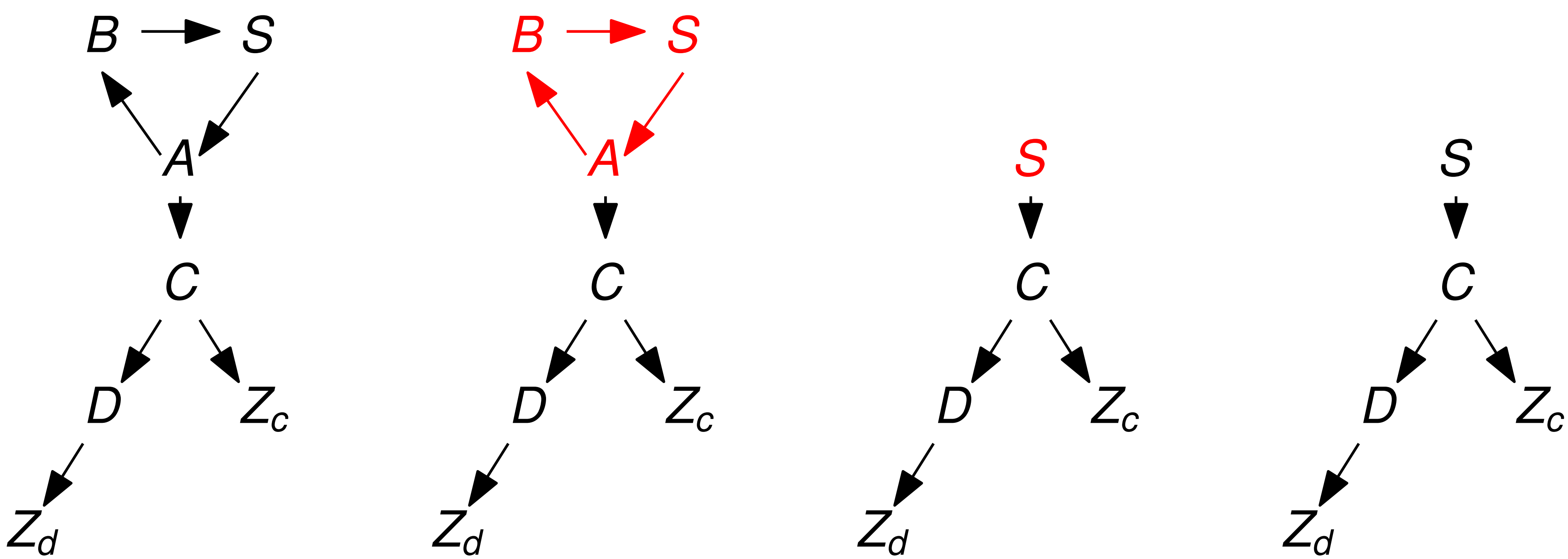
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.



Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.

$$S \to A \mid Z_a B \mid Z_a C$$

$$B \to S \mid B Z_a$$

$$D \to Z_d \mid Z_d E$$

$$A \to B \mid C \mid Z_c F$$

$$C \to D \mid Z_c$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

$$E \to DD \quad F \to A Z_d$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.

$$S \to A \mid Z_a B \mid Z_a C$$

$$B \to S \mid B Z_a$$

$$D \to Z_d \mid Z_d E$$

$$A \to B \mid C \mid Z_c F$$

$$C \to D \mid Z_c$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

$$E \to DD \quad F \to AZ_d$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.

$$S \to A \mid Z_aB \mid Z_aC$$
$$B \to S \mid BZ_a$$
$$D \to Z_d \mid Z_dE$$
$$A \to B \mid C \mid Z_cF$$
$$C \to D \mid Z_c$$
$$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$$
$$E \to DD \quad F \to AZ_d$$

$$S \to Z_aS \mid Z_aC \mid SZ_a \mid C \mid Z_cF$$
$$D \to Z_d \mid Z_dE$$
$$C \to D \mid Z_c$$
$$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$$
$$E \to DD \quad F \to SZ_d$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.

$$S \to A \mid Z_aB \mid Z_aC$$
$$B \to S \mid BZ_a$$
$$D \to Z_d \mid Z_dE$$
$$A \to B \mid C \mid Z_cF$$
$$C \to D \mid Z_c$$
$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$
$$E \to DD \qquad F \to AZ_d$$

$$S \to Z_aS \mid Z_aC \mid SZ_a \mid C \mid Z_cF$$
$$D \to Z_d \mid Z_dE$$
$$C \to D \mid Z_c$$
$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$
$$E \to DD \qquad F \to SZ_d$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere Kreis $B \to S \to A \to B$.

$S \to A \mid Z_aB \mid Z_aC$

$B \to S \mid BZ_a$

$D \to Z_d \mid Z_dE$

$A \to B \mid C \mid Z_cF$

$C \to D \mid Z_c$

$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$

$E \to DD \quad F \to AZ_d$

$S \to Z_aS \mid Z_aC \mid SZ_a \mid C \mid Z_cF$

$D \to Z_d \mid Z_dE$

$C \to D \mid Z_c$

$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$

$E \to DD \quad F \to SZ_d$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere verbleibende Kettenregeln: $A \to B$.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere verbleibende Kettenregeln: $A \to B$.

(a) Topologische Sortierung: S, C, D, Z_c, Z_d,

(b) Keine Kettenregeln mit linker Seite Z_d und Z_c,

(c) Ersetze Kettenregeln mit linker Seite D,

(d) Ersetze Kettenregeln mit linker Seite C,

(e) Ersetze Kettenregeln mit linker Seite S.

$$\begin{aligned}
S &\to Z_aS \mid Z_aC \\
&\quad\;\, SZ_a \mid C \mid Z_cF \\
D &\to Z_d \mid Z_dE \\
C &\to D \mid Z_c \\
Z_a &\to \mathtt{a} \\
Z_c &\to \mathtt{c} \\
Z_d &\to \mathtt{d} \\
E &\to DD \\
F &\to SZ_d
\end{aligned}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere verbleibende Kettenregeln: $A \to B$.

$$S \to Z_aS \mid Z_aC \mid SZ_a \mid C \mid Z_cF$$
$$D \to Z_d \mid Z_dE$$
$$C \to D \mid Z_c$$
$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$
$$E \to DD \quad F \to SZ_d$$

$$S \to Z_aS \mid Z_aC \mid SZ_a \mid \mathtt{d} \mid Z_dE \mid \mathtt{c} \mid Z_cF$$
$$D \to \mathtt{d} \mid Z_dE$$
$$C \to \mathtt{d} \mid Z_dE \mid \mathtt{c}$$
$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$
$$E \to DD \quad F \to SZ_d$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere verbleibende Kettenregeln: $A \to B$.

$S \to Z_aS \mid Z_aC \mid SZ_a \mid C \mid Z_cF$	$S \to Z_aS \mid Z_aC \mid SZ_a \mid \mathtt{d} \mid Z_dE \mid \mathtt{c} \mid Z_cF$
$D \to Z_d \mid Z_dE$	$D \to \mathtt{d} \mid Z_dE$
$C \to D \mid Z_c$	$C \to \mathtt{d} \mid Z_dE \mid \mathtt{c}$
$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$	$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$
$E \to DD \quad F \to SZ_d$	$E \to DD \quad F \to SZ_d$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky-Normalform

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

4. Schritt: Eliminiere verbleibende Kettenregeln: $A \to B$.

$S \to Z_aS \mid Z_aC \mid SZ_a \mid C \mid Z_cF$	$S \to Z_aS \mid Z_aC \mid SZ_a \mid \mathtt{d} \mid Z_dE \mid \mathtt{c} \mid Z_cF$
$D \to Z_d \mid Z_dE$	$D \to \mathtt{d} \mid Z_dE$
$C \to D \mid Z_c$	$C \to \mathtt{d} \mid Z_dE \mid \mathtt{c}$
$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$	$Z_a \to \mathtt{a} \quad Z_c \to \mathtt{c} \quad Z_d \to \mathtt{d}$
$E \to DD \quad F \to SZ_d$	$E \to DD \quad F \to SZ_d$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Der Cocke-Younger-Kasami Algorithmus

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, $V = \{S, A, B, C, D\}$ und R:

$$S \to A \mid \mathtt{a}B \mid \mathtt{a}C, \quad B \to S \mid B\mathtt{a}, \quad D \to \mathtt{d} \mid \mathtt{d}DD,$$
$$A \to B \mid C \mid \mathtt{c}A\mathtt{d}, \quad C \to D \mid \mathtt{c}.$$

b) Prüfen Sie, ob das Wort $\mathtt{addd}$ in $L(G)$ liegt. Verwenden Sie dafür den CYK-Algorithmus.

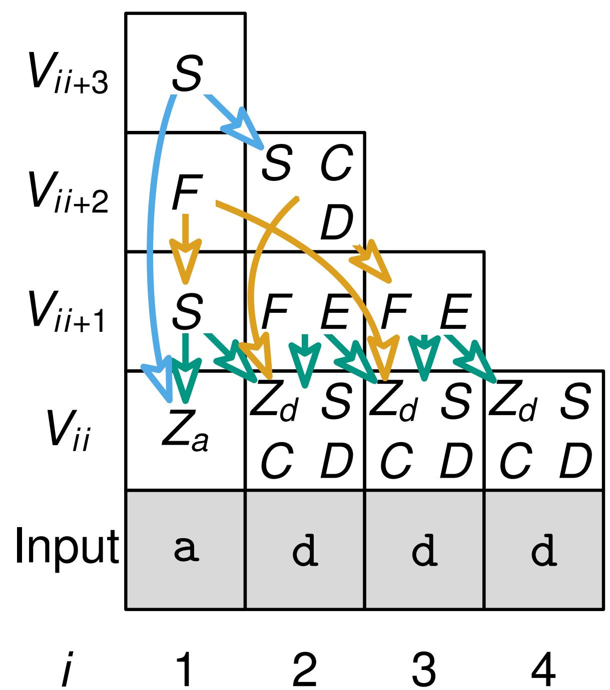
$$S \to Z_aS \mid Z_aC \mid SZ_a \mid \mathtt{d} \mid Z_dE \mid \mathtt{c} \mid Z_cF$$

$$D \to \mathtt{d} \mid Z_dE$$

$$C \to \mathtt{d} \mid Z_dE \mid \mathtt{c}$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

$$E \to DD \qquad F \to SZ_d$$



Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Der Cocke-Younger-Kasami Algorithmus

$$S \to Z_a S \mid Z_a C \mid S Z_a \mid \mathtt{d} \mid Z_d E \mid \mathtt{c} \mid Z_c F$$

$$D \to \mathtt{d} \mid Z_d E$$

Chomsky-Normalform

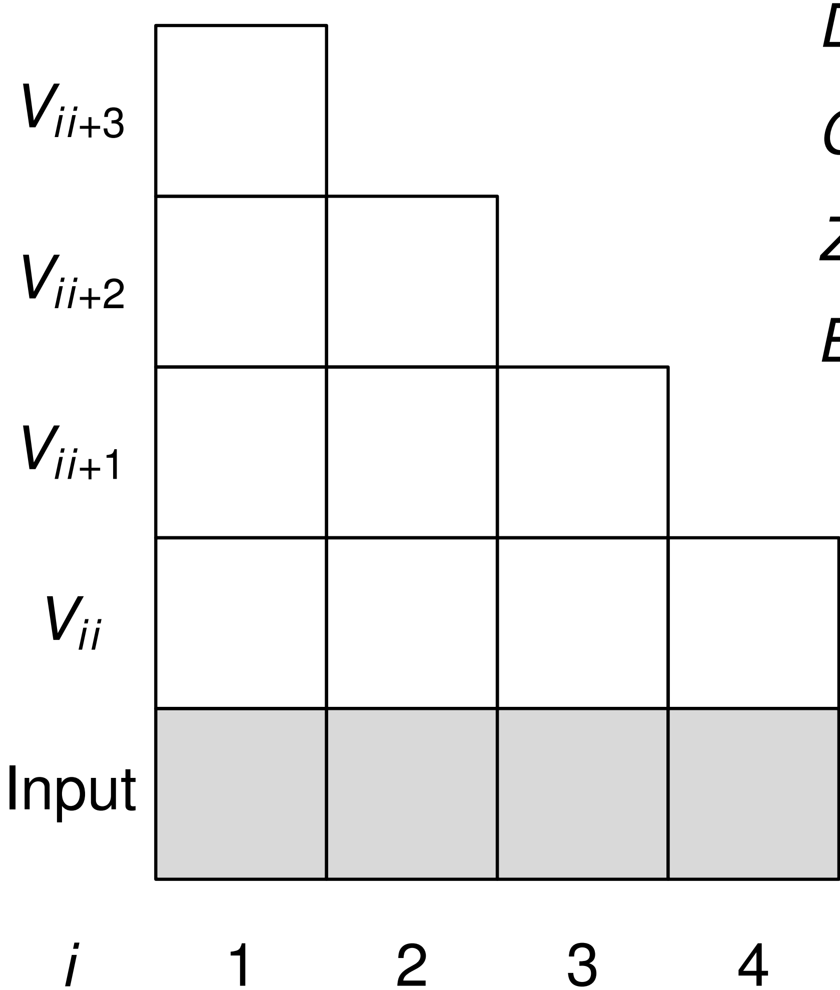
$$C \to \mathtt{d} \mid Z_d E \mid \mathtt{c}$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

$$E \to DD \quad F \to S Z_d$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Der Cocke-Younger-Kasami Algorithmus



$$S \to Z_aS \mid Z_aC \mid SZ_a \mid \mathtt{d} \mid Z_dE \mid \mathtt{c} \mid Z_cF$$

$$D \to \mathtt{d} \mid Z_dE$$

$$C \to \mathtt{d} \mid Z_dE \mid \mathtt{c}$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

$$E \to DD \quad F \to SZ_d$$

Chomsky-Normalform

① Input $w = w_1 \dots w_n$ aus Σ^*.

② Variable $A \in V$ ist in V_{ii+j} gdw. $A \overset{*}{\to} w_i \dots w_{i+j}$.

③ w ist in $L(G)$ gdw. $S \in V_{1n}$.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Der Cocke-Younger-Kasami Algorithmus

$S \to Z_a S \mid Z_a C \mid S Z_a \mid \mathtt{d} \mid Z_d E \mid \mathtt{c} \mid Z_c F$

$D \to \mathtt{d} \mid Z_d E$

Chomsky-Normalform

$C \to \mathtt{d} \mid Z_d E \mid \mathtt{c}$

$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$

$E \to DD \quad F \to S Z_d$

V_{ii+3}				
V_{ii+2}				
V_{ii+1}				
V_{ii}				
Input	a	d	d	d
i	1	2	3	4

① Input $w = w_1 \dots w_n$ aus Σ^*.

② Variable $A \in V$ ist in V_{ii+j} gdw. $A \xrightarrow{*} w_i \dots w_{i+j}$.

③ w ist in $L(G)$ gdw. $S \in V_{1n}$.

Der Cocke-Younger-Kasami Algorithmus

$$S \to Z_aS \mid Z_aC \mid SZ_a \mid \mathtt{d} \mid Z_dE \mid \mathtt{c} \mid Z_cF$$

$$D \to \mathtt{d} \mid Z_dE$$

Chomsky-Normalform

$$C \to \mathtt{d} \mid Z_dE \mid \mathtt{c}$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

$$E \to DD \qquad F \to SZ_d$$

V_{ii+3}				
V_{ii+2}				
V_{ii+1}				
V_{ii}	Z_a	Z_d S C D	Z_d S C D	Z_d S C D
Input	a	d	d	d
i	1	2	3	4

① Input $w = w_1 \ldots w_n$ aus Σ^*.

② Variable $A \in V$ ist in V_{ii+j} gdw. $A \xrightarrow{*} w_i \ldots w_{i+j}$.

③ w ist in $L(G)$ gdw. $S \in V_{1n}$.

Der Cocke-Younger-Kasami Algorithmus

$S \to Z_a S \mid Z_a C \mid S Z_a \mid \mathtt{d} \mid Z_d E \mid \mathtt{c} \mid Z_c F$

$D \to \mathtt{d} \mid Z_d E$

$C \to \mathtt{d} \mid Z_d E \mid \mathtt{c}$

$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$

$E \to DD \qquad F \to S Z_d$

Chomsky-Normalform

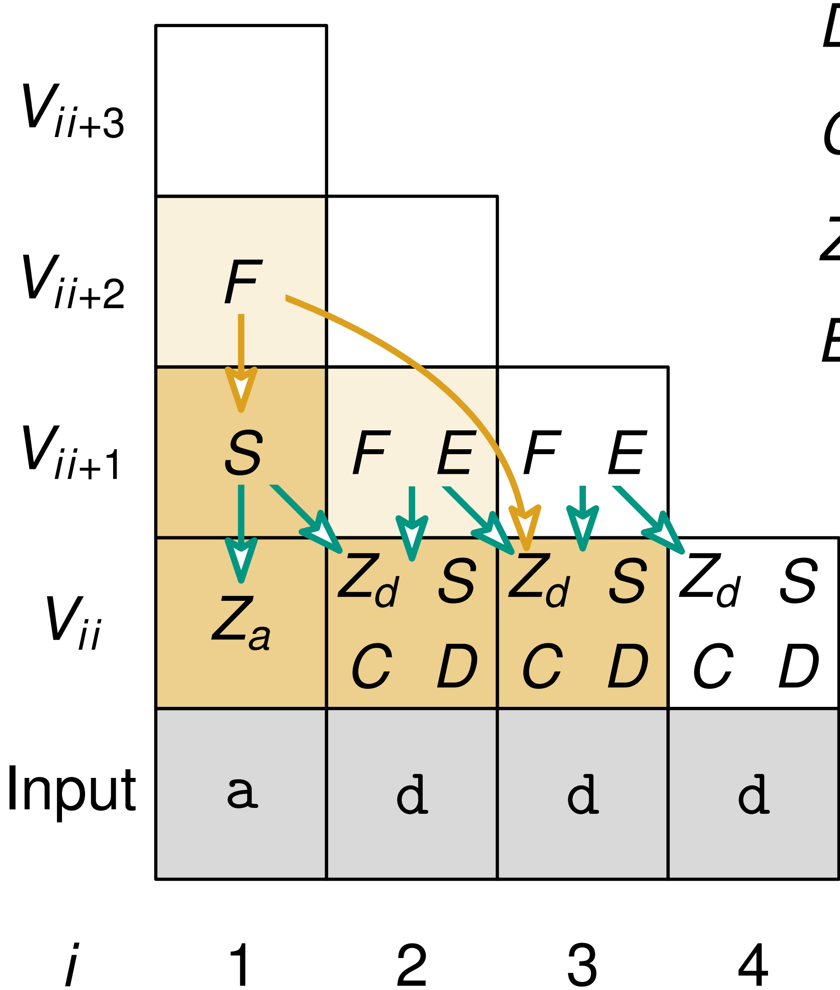
V_{ii+3}				
V_{ii+2}				
V_{ii+1}	S	F E	F E	
V_{ii}	Z_a	Z_d S C D	Z_d S C D	Z_d S C D
Input	a	d	d	d
i	1	2	3	4

① Input $w = w_1 \dots w_n$ aus Σ^*.

② Variable $A \in V$ ist in V_{ii+j} gdw. $A \xrightarrow{*} w_i \dots w_{i+j}$.

③ w ist in $L(G)$ gdw. $S \in V_{1n}$.

Der Cocke-Younger-Kasami Algorithmus



$$S \to Z_aS \mid Z_aC \mid SZ_a \mid \mathtt{d} \mid Z_dE \mid \mathtt{c} \mid Z_cF$$

$$D \to \mathtt{d} \mid Z_dE$$

Chomsky-Normalform

$$C \to \mathtt{d} \mid Z_dE \mid \mathtt{c}$$

$$Z_a \to \mathtt{a} \qquad Z_c \to \mathtt{c} \qquad Z_d \to \mathtt{d}$$

$$E \to DD \qquad F \to SZ_d$$

① Input $w = w_1 \dots w_n$ aus Σ^*.

② Variable $A \in V$ ist in V_{ii+j} gdw. $A \stackrel{*}{\to} w_i \dots w_{i+j}$.

③ w ist in $L(G)$ gdw. $S \in V_{1n}$.

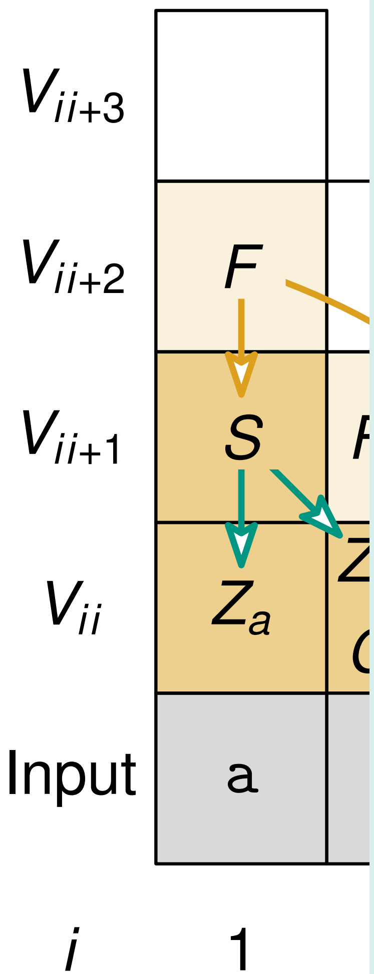
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Der Cocke-Younger-Kasami Algorithmus

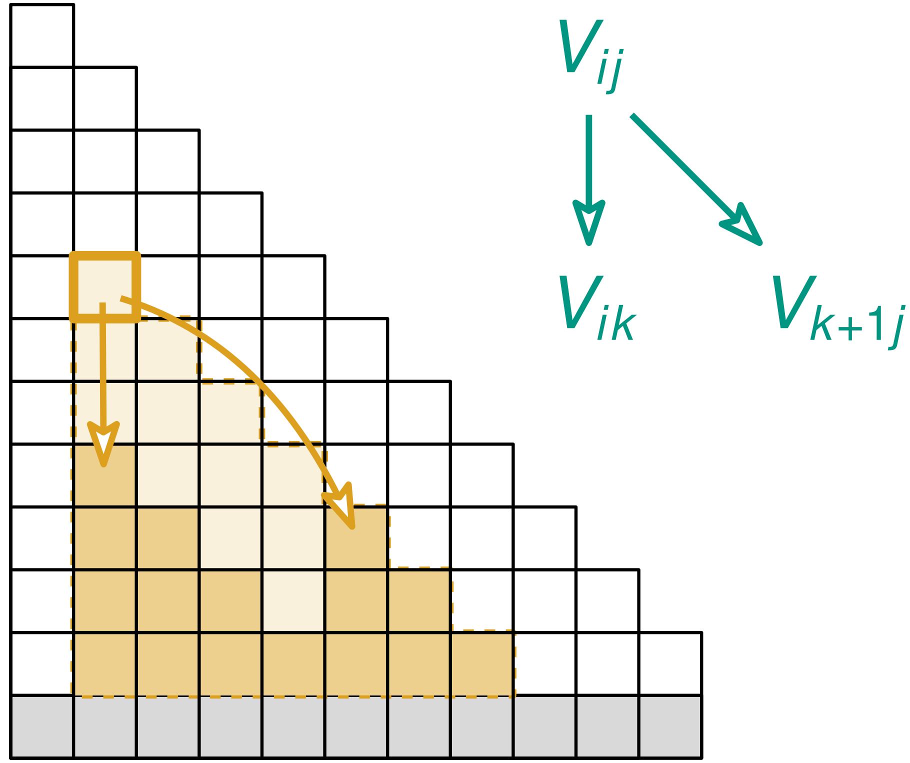
$Z_d E \mid c \mid Z_c F$

y-Normalform

$\to$ d



Der allgemeine Fall



aus Σ^*.

in V_{ii+j}

. w_{i+j}.

$S \in V_{1n}$.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Der Cocke-Younger-Kasami Algorithmus

V_{ii+3}				
V_{ii+2}	F	$S\ C\ D$		
V_{ii+1}	S	$F\ E$	$F\ E$	
V_{ii}	Z_a	$Z_d\ S\ C\ D$	$Z_d\ S\ C\ D$	$Z_d\ S\ C\ D$
Input	a	d	d	d
i	1	2	3	4

$$S \rightarrow Z_aS \mid Z_aC \mid SZ_a \mid \mathtt{d} \mid Z_dE \mid \mathtt{c} \mid Z_cF$$

$$D \rightarrow \mathtt{d} \mid Z_dE$$

$$C \rightarrow \mathtt{d} \mid Z_dE \mid \mathtt{c}$$

$$Z_a \rightarrow \mathtt{a} \qquad Z_c \rightarrow \mathtt{c} \qquad Z_d \rightarrow \mathtt{d}$$

$$E \rightarrow DD \quad F \rightarrow SZ_d$$

Chomsky-Normalform

① Input $w = w_1 \ldots w_n$ aus Σ^*.

② Variable $A \in V$ ist in V_{ii+j} gdw. $A \overset{*}{\rightarrow} w_i \ldots w_{i+j}$.

③ w ist in $L(G)$ gdw. $S \in V_{1n}$.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Der Cocke-Younger-Kasami Algorithmus

V_{ii+3}	S			
V_{ii+2}	F	$S\ C$ D		
V_{ii+1}	S	$F\ E$	$F\ E$	
V_{ii}	Z_a	$Z_d\ S$ $C\ D$	$Z_d\ S$ $C\ D$	$Z_d\ S$ $C\ D$
Input	a	d	d	d
i	1	2	3	4

$S \to Z_aS \mid Z_aC \mid SZ_a \mid \mathtt{d} \mid Z_dE \mid \mathtt{c} \mid Z_cF$

$D \to \mathtt{d} \mid Z_dE$

$C \to \mathtt{d} \mid Z_dE \mid \mathtt{c}$

$Z_a \to \mathtt{a}$ $\quad Z_c \to \mathtt{c}$ $\quad Z_d \to \mathtt{d}$

$E \to DD$ $\quad F \to SZ_d$

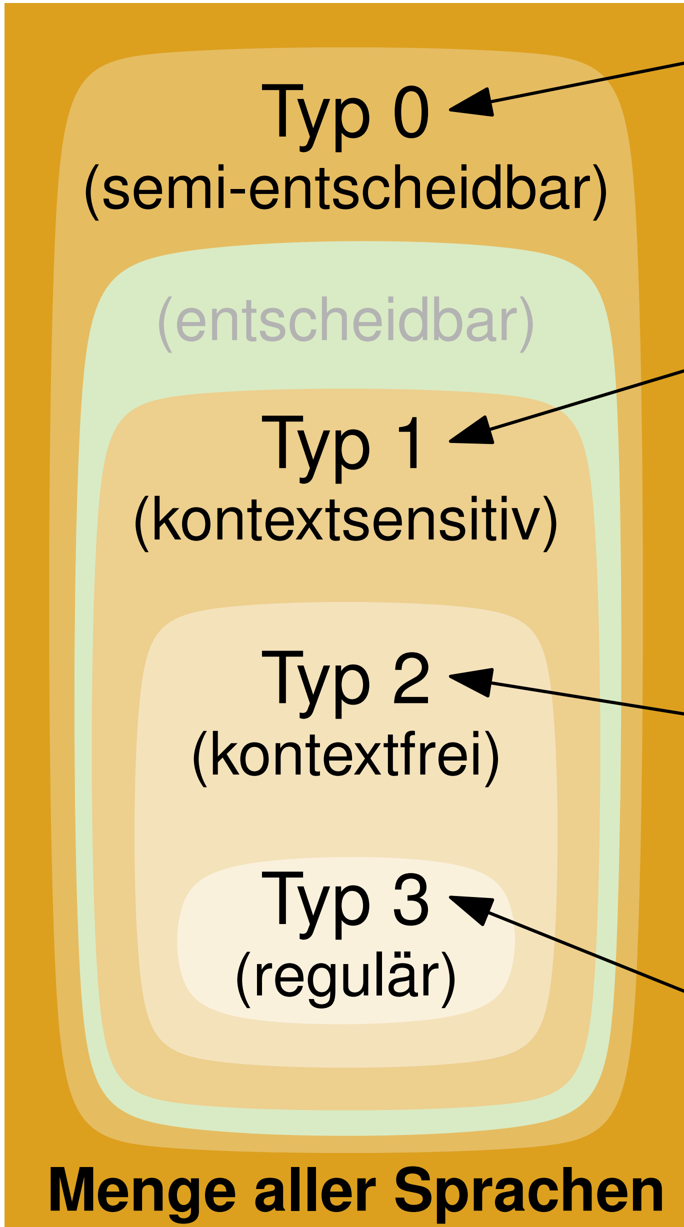
Chomsky-Normalform

① Input $w = w_1 \ldots w_n$ aus Σ^*.

② Variable $A \in V$ ist in V_{ii+j} gdw. $A \overset{*}{\to} w_i \ldots w_{i+j}$.

③ w ist in $L(G)$ gdw. $S \in V_{1n}$.

Chomsky Hierarchie



Chomsky-0-Gram.

$G = (\Sigma, V, S, R)$

R beliebig

NTM akzeptiert L
DTM akzeptiert L

Chomsky-1-Gram.

$u \to w, |u| \leq |w|$

$u \in V^+, S \notin w$

$S \to \varepsilon$

NTM mit Platzbedarf n erkennt Wörter d. Länge n in L

$\Rightarrow \mathcal{NTAPE}(n)$

Chomsky-2-Gram.

$A \to w, A \in V$

w beliebig

CYK-Alg. erkennt L in polynom. Zeit

$\Rightarrow$ Chomsky-Normalform

Chomsky-3-Gram.

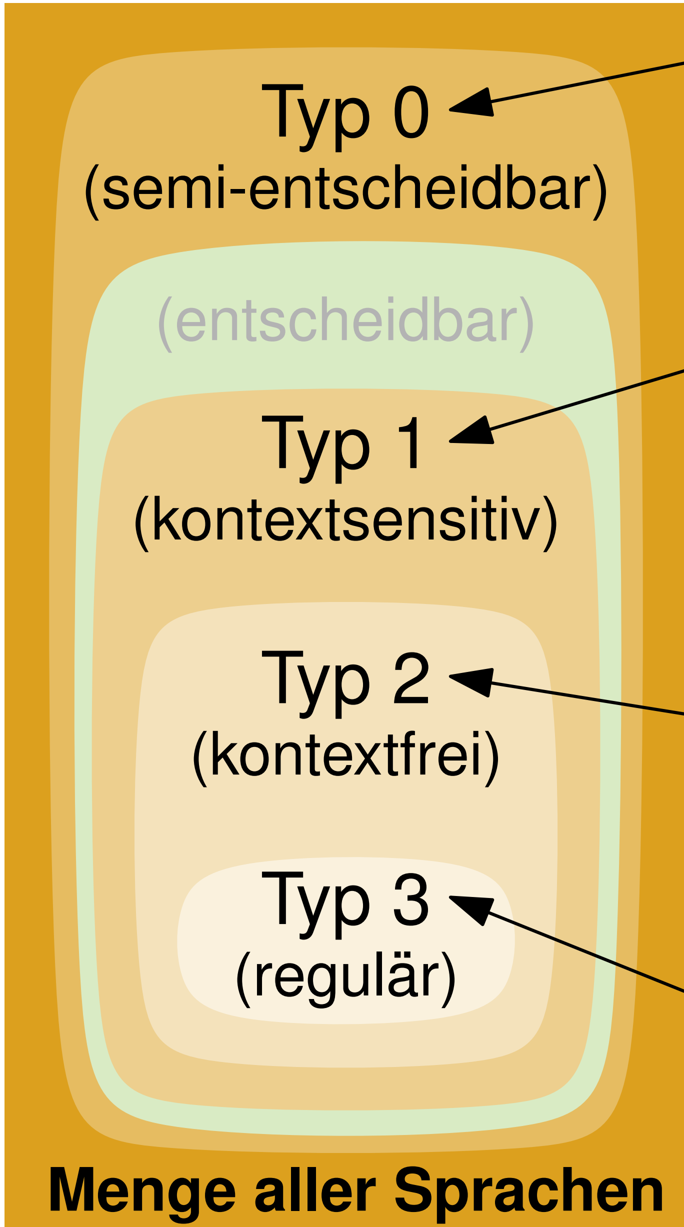
$A \to w, A \in V$

$w \in \{\varepsilon\} \cup \Sigma \cdot V$

NEA erkennt L
DEA erkennt L

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Chomsky Hierarchie



Chomsky-0-Gram.

$G = (\Sigma, V, S, R)$

R beliebig

$\Leftrightarrow$ NTM akzeptiert L / DTM akzeptiert L

Chomsky-1-Gram.

$u \to w, |u| \leq |w|$

$u \in V^+, S \notin w$

$S \to \varepsilon$

$\Leftrightarrow$ NTM mit Platzbedarf n erkennt Wörter d. Länge n in L

$\Rightarrow \mathcal{NTAPE}(n)$

Chomsky-2-Gram.

$A \to w, A \in V$

w beliebig

$\Leftrightarrow$ CYK-Alg. erkennt L in polynom. Zeit

$\Rightarrow$ Chomsky-Normalform

Chomsky-3-Gram.

$A \to w, A \in V$

$w \in \{\varepsilon\} \cup \Sigma \cdot V$

$\Leftrightarrow$ NEA erkennt L / DEA erkennt L

NEA aus Chomsky-3-Grammatik

Gegeben sei die Grammatik $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ und $V = \{X, Y, Z, S\}$, welche die Sprache L erzeugt. R sei durch die folgenden Ableitgungsregeln gegeben.

$$\begin{aligned} S &\rightarrow \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X \\ X &\rightarrow \mathtt{a}S \mid \mathtt{b}Y \\ Y &\rightarrow \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z \\ Z &\rightarrow \mathtt{b}Z \mid \mathtt{a}S \end{aligned}$$

Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

NEA aus Chomsky-3-Grammatik

Gegeben: $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $V = \{X, Y, Z, S\}$ und $R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, X \to \mathtt{a}S \mid \mathtt{b}Y, Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, Z \to \mathtt{b}Z \mid \mathtt{a}S\}$.
Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

NEA aus Chomsky-3-Grammatik

Gegeben: $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $V = \{X, Y, Z, S\}$ und $R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, X \to \mathtt{a}S \mid \mathtt{b}Y, Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, Z \to \mathtt{b}Z \mid \mathtt{a}S\}$.
Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

$$V = \{X, Y, Z, S\}$$

$$\begin{aligned} R = \{S &\to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, \\ X &\to \mathtt{a}S \mid \mathtt{b}Y, \\ Y &\to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, \\ Z &\to \mathtt{b}Z \mid \mathtt{a}S\}. \end{aligned}$$

NEA aus Chomsky-3-Grammatik

Gegeben: $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $V = \{X, Y, Z, S\}$ und $R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, X \to \mathtt{a}S \mid \mathtt{b}Y, Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, Z \to \mathtt{b}Z \mid \mathtt{a}S\}$.
Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

$V = \{X, Y, Z, S\}$

$$\begin{aligned} R = \{&S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, \\ &X \to \mathtt{a}S \mid \mathtt{b}Y, \\ &Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, \\ &Z \to \mathtt{b}Z \mid \mathtt{a}S\}. \end{aligned}$$

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

NEA aus Chomsky-3-Grammatik

Gegeben: $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $V = \{X, Y, Z, S\}$ und $R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, X \to \mathtt{a}S \mid \mathtt{b}Y, Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, Z \to \mathtt{b}Z \mid \mathtt{a}S\}$.
Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

$V = \{X, Y, Z, S\}$

$$\begin{aligned} R = \{S &\to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, \\ X &\to \mathtt{a}S \mid \mathtt{b}Y, \\ Y &\to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, \\ Z &\to \mathtt{b}Z \mid \mathtt{a}S\}. \end{aligned}$$

NEA aus Chomsky-3-Grammatik

Gegeben: $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $V = \{X, Y, Z, S\}$ und $R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, X \to \mathtt{a}S \mid \mathtt{b}Y, Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, Z \to \mathtt{b}Z \mid \mathtt{a}S\}$.
Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

$V = \{X, Y, Z, S\}$

$$R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X,$$
$$X \to \mathtt{a}S \mid \mathtt{b}Y,$$
$$Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z,$$
$$Z \to \mathtt{b}Z \mid \mathtt{a}S\}.$$

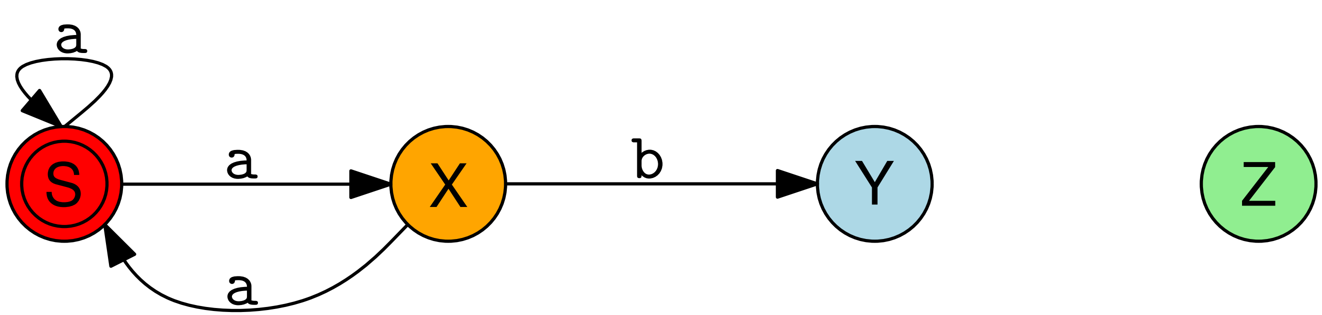
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

NEA aus Chomsky-3-Grammatik

Gegeben: $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $V = \{X, Y, Z, S\}$ und $R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, X \to \mathtt{a}S \mid \mathtt{b}Y, Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, Z \to \mathtt{b}Z \mid \mathtt{a}S\}$.
Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

$V = \{X, Y, Z, S\}$

$$R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X,\\ X \to \mathtt{a}S \mid \mathtt{b}Y,\\ Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z,\\ Z \to \mathtt{b}Z \mid \mathtt{a}S\}.$$



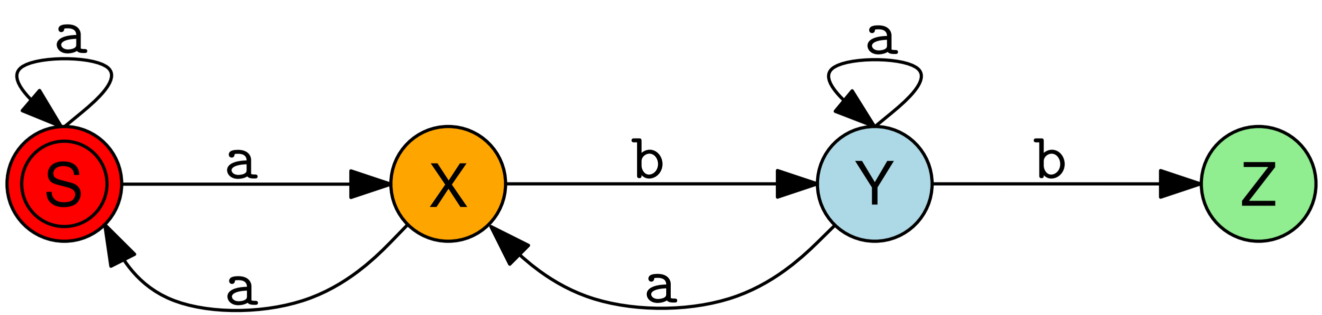
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

NEA aus Chomsky-3-Grammatik

Gegeben: $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $V = \{X, Y, Z, S\}$ und $R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, X \to \mathtt{a}S \mid \mathtt{b}Y, Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, Z \to \mathtt{b}Z \mid \mathtt{a}S\}$.
Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

$V = \{X, Y, Z, S\}$

$$\begin{aligned} R = \{&S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X,\\ &X \to \mathtt{a}S \mid \mathtt{b}Y,\\ &Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z,\\ &Z \to \mathtt{b}Z \mid \mathtt{a}S\}. \end{aligned}$$



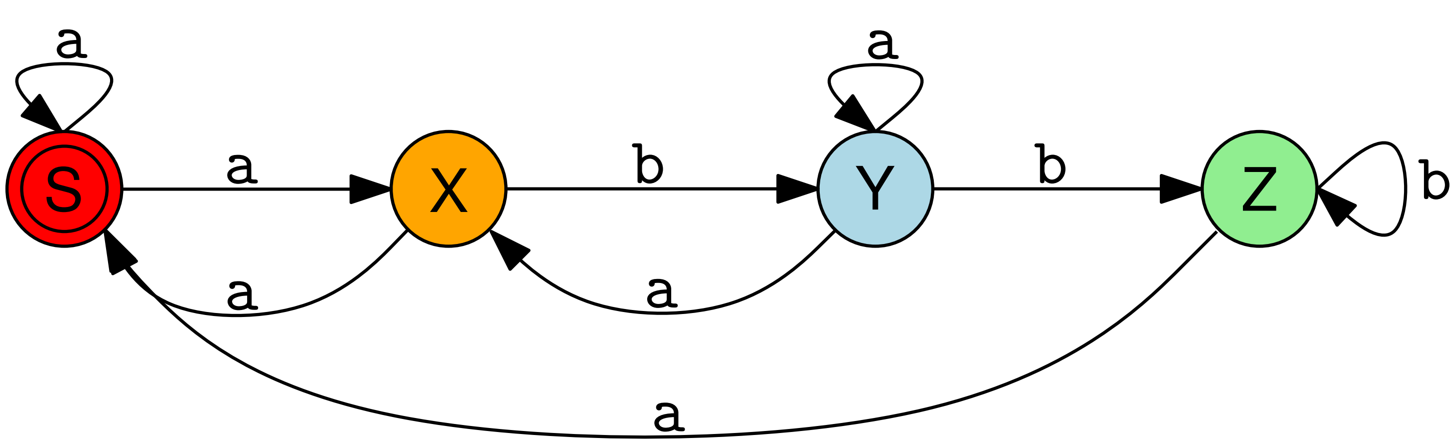
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

NEA aus Chomsky-3-Grammatik

Gegeben: $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $V = \{X, Y, Z, S\}$ und $R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, X \to \mathtt{a}S \mid \mathtt{b}Y, Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, Z \to \mathtt{b}Z \mid \mathtt{a}S\}$.
Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

$V = \{X, Y, Z, S\}$

$$\begin{aligned} R = \{ & S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, \\ & X \to \mathtt{a}S \mid \mathtt{b}Y, \\ & Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, \\ & Z \to \mathtt{b}Z \mid \mathtt{a}S\}. \end{aligned}$$



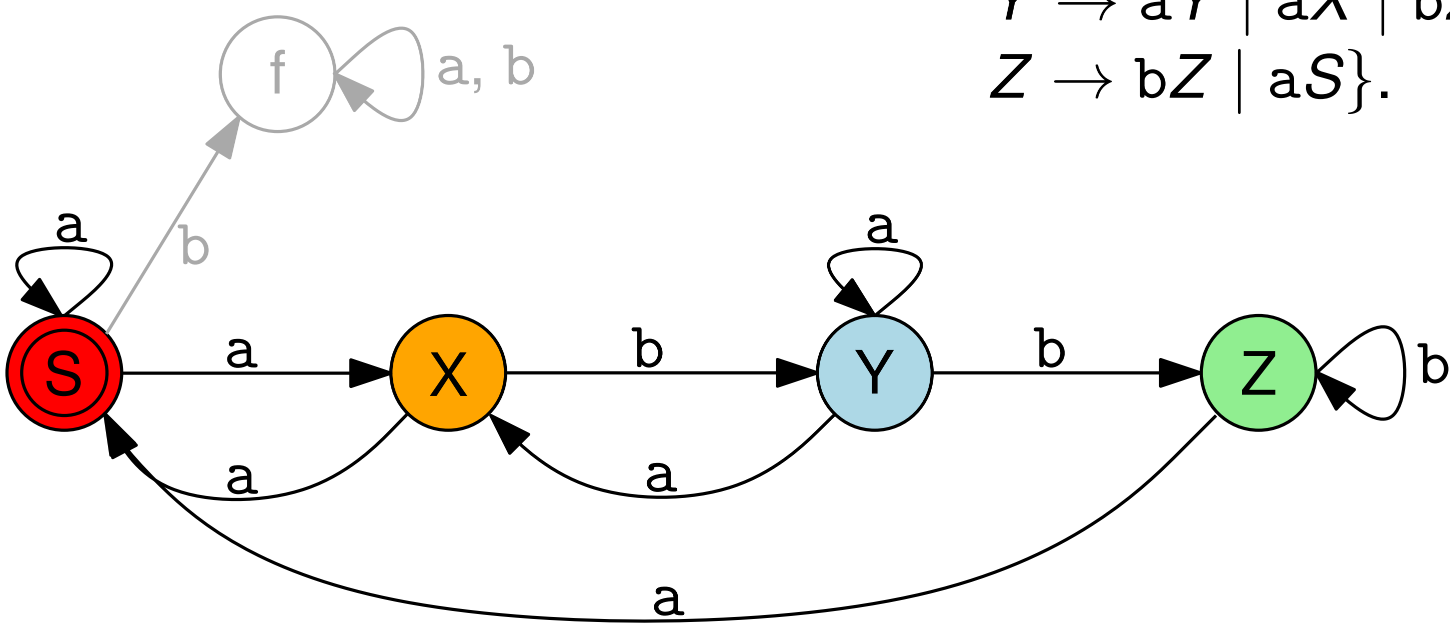
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

NEA aus Chomsky-3-Grammatik

Gegeben: $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $V = \{X, Y, Z, S\}$ und $R = \{S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, X \to \mathtt{a}S \mid \mathtt{b}Y, Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, Z \to \mathtt{b}Z \mid \mathtt{a}S\}$.
Geben Sie einen nichtdeterministischen endlichen Automaten $\mathcal{A}$ an, der L erkennt.

$V = \{X, Y, Z, S\}$

$$\begin{aligned} R = \{ & S \to \varepsilon \mid \mathtt{a}S \mid \mathtt{a}X, \\ & X \to \mathtt{a}S \mid \mathtt{b}Y, \\ & Y \to \mathtt{a}Y \mid \mathtt{a}X \mid \mathtt{b}Z, \\ & Z \to \mathtt{b}Z \mid \mathtt{a}S\}. \end{aligned}$$



Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Eindeutige und mehrdeutige Grammatiken

Gegeben sei die Grammatik $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\texttt{0}, \texttt{1}, \texttt{2}, \texttt{3}, \texttt{4}, \texttt{5}, \texttt{6}, \texttt{7}, \texttt{8}, \texttt{9}, \texttt{+}, \texttt{-}, \texttt{*}\}$ und $V = \{S, Z\}$. R :

$$\begin{array}{lcl} S & \rightarrow & S\texttt{+}S \mid S\texttt{-}S \mid S\texttt{*}S \mid Z, \\ Z & \rightarrow & \texttt{0} \mid \texttt{1} \mid \texttt{2} \mid \texttt{3} \mid \texttt{4} \mid \texttt{5} \mid \texttt{6} \mid \texttt{7} \mid \texttt{8} \mid \texttt{9} \mid \texttt{0}Z \mid \texttt{1}Z \mid \texttt{2}Z \mid \texttt{3}Z \mid \texttt{4}Z \mid \texttt{5}Z \mid \\ & & \texttt{6}Z \mid \texttt{7}Z \mid \texttt{8}Z \mid \texttt{9}Z\ . \end{array}$$

Bestimmen Sie einen Syntaxbaum des Wortes `211-42+10*4`. Ist die Grammatik eindeutig oder inhärent mehrdeutig? Begründen Sie Ihre Antwort.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Eindeutige und mehrdeutige Grammatiken

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, *\}$, $V = \{S, Z\}$, R :

$$\begin{aligned} S &\rightarrow S{+}S \mid S{-}S \mid S{*}S \mid Z, \\ Z &\rightarrow 0 \mid 1 \mid \ldots \mid 9 \mid 0Z \mid 1Z \mid \ldots \mid 9Z \,. \end{aligned}$$

Ist G eindeutig oder inhärent mehrdeutig?

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

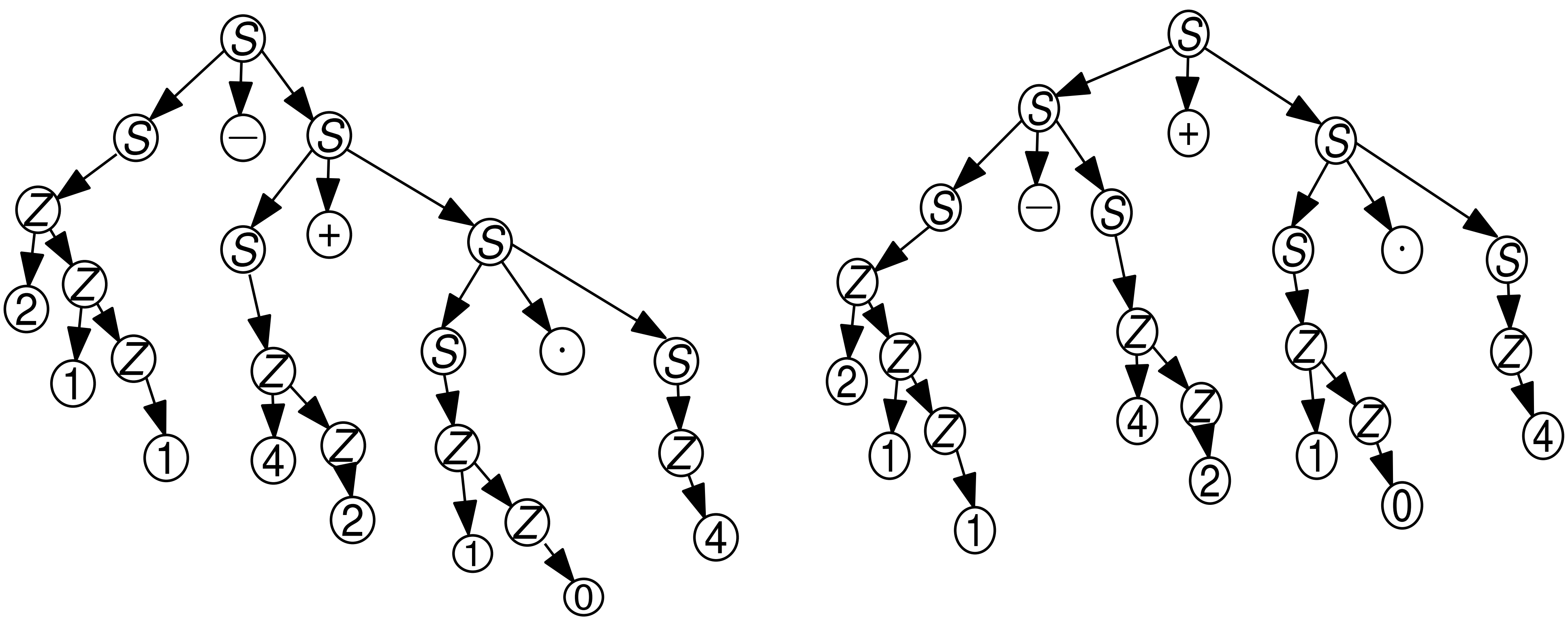
Eindeutige und mehrdeutige Grammatiken

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, *\}$, $V = \{S, Z\}$, R :

$$S \rightarrow S{+}S \mid S{-}S \mid S{*}S \mid Z,$$

$$Z \rightarrow 0 \mid 1 \mid \ldots \mid 9 \mid 0Z \mid 1Z \mid \ldots \mid 9Z\ .$$

Ist G eindeutig oder inhärent mehrdeutig?



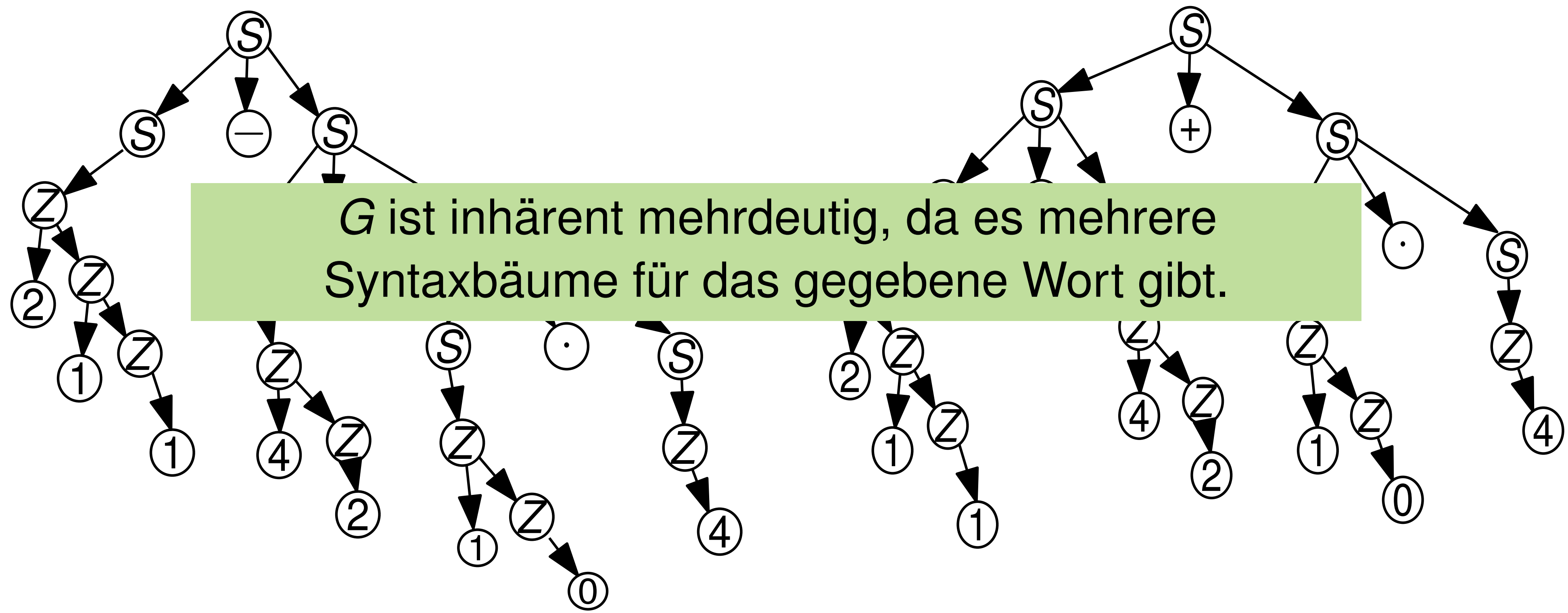
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Eindeutige und mehrdeutige Grammatiken

$G = (\Sigma, V, S, R)$ mit $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, *\}$, $V = \{S, Z\}$, R :

$$S \rightarrow S{+}S \mid S{-}S \mid S{*}S \mid Z,$$
$$Z \rightarrow 0 \mid 1 \mid \ldots \mid 9 \mid 0Z \mid 1Z \mid \ldots \mid 9Z\ .$$

Ist G eindeutig oder inhärent mehrdeutig?



G ist inhärent mehrdeutig, da es mehrere Syntaxbäume für das gegebene Wort gibt.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik