

Kapitel 6

Kontextfreie Sprachen

Abschnitt 6.1

Kontextfreie Grammatiken

Die Syntax komplexer formaler Sprachen wie Programmiersprachen oder Spezifikationssprachen ist oft induktiv definiert.

Die Syntax komplexer formaler Sprachen wie Programmiersprachen oder Spezifikationssprachen ist oft induktiv definiert.

Solche induktiven Definitionen lassen sich kompakt und präzise durch **Grammatiken** beschreiben. In der Regel handelt es sich dabei um **kontextfreie** Grammatiken.

In der Praxis werden kontextfreie Grammatiken oft in **Backus-Naur Form (BNF)** angegeben.

In der Praxis werden kontextfreie Grammatiken oft in **Backus-Naur Form (BNF)** angegeben.

Beispiel 6.1

Folgende Grammatik in BNF beschreibt unsere induktiv definierten Mengen BIN der Binärzahlen (vgl. Beispiel 3.6) und und ART der arithmetischen Terme (vgl. Beispiel 3.20):

```
<digit>    ::= "0" | "1"
<bstring>  ::= "" | <digit> <bstring>
<bin>      ::= "0" | "1" <bstring>
<op2>      ::= "+" | "*" | "-"
<art>      ::= <bin>
              | "-" <art>
              | "(" <art> <op2> <art> ")"
```

Kontextfreie Grammatiken (Syntax)

Definition 6.2

Eine **kontextfreie Grammatik** ist ein Quadrupel

$$(N, \Sigma, P, S),$$

Kontextfreie Grammatiken (Syntax)

Definition 6.2

Eine **kontextfreie Grammatik** ist ein Quadrupel

$$(N, \Sigma, P, S),$$

bestehend aus

- ▶ einer endlichen Menge N , deren Elemente wir als **Nichtterminalsymbole** bezeichnen,

Kontextfreie Grammatiken (Syntax)

Definition 6.2

Eine **kontextfreie Grammatik** ist ein Quadrupel

$$(N, \Sigma, P, S),$$

bestehend aus

- ▶ einer endlichen Menge N , deren Elemente wir als **Nichtterminalsymbole** bezeichnen,
- ▶ einer zu N disjunkten endlichen Menge Σ , dem **Terminalalphabet**, dessen Elemente wir als **Terminalsymbole** bezeichnen,

Kontextfreie Grammatiken (Syntax)

Definition 6.2

Eine **kontextfreie Grammatik** ist ein Quadrupel

$$(N, \Sigma, P, S),$$

bestehend aus

- ▶ einer endlichen Menge N , deren Elemente wir als **Nichtterminalsymbole** bezeichnen,
- ▶ einer zu N disjunkten endlichen Menge Σ , dem **Terminalalphabet**, dessen Elemente wir als **Terminalsymbole** bezeichnen,
- ▶ einer endlichen Menge P von **Regeln** (oder **Produktionen**) der Form $A \rightarrow \alpha$ mit $A \in N$ und $\alpha \in (N \cup \Sigma)^*$,

Kontextfreie Grammatiken (Syntax)

Definition 6.2

Eine **kontextfreie Grammatik** ist ein Quadrupel

$$(N, \Sigma, P, S),$$

bestehend aus

- ▶ einer endlichen Menge N , deren Elemente wir als **Nichtterminalsymbole** bezeichnen,
- ▶ einer zu N disjunkten endlichen Menge Σ , dem **Terminalalphabet**, dessen Elemente wir als **Terminalsymbole** bezeichnen,
- ▶ einer endlichen Menge P von **Regeln** (oder **Produktionen**) der Form $A \rightarrow \alpha$ mit $A \in N$ und $\alpha \in (N \cup \Sigma)^*$,
- ▶ einem **Startsymbol** $S \in N$.

Beispiel 6.3

Wir betrachten noch einmal die Grammatik für die Menge $\text{ART} \subseteq \Sigma^*$, die in BNF wie folgt aussah (vgl. Beispiel 6.1):

```
<digit>    ::= "0" | "1"
<bstring>  ::= "" | <digit> <bstring>
<bin>      ::= "0" | "1" <bstring>
<op2>      ::= "+" | "*" | "-"
<art>      ::= <bin> | "-" <art>
              | "(" <art> <op2> <art> ")"
```

Beispiel 6.3

Wir betrachten noch einmal die Grammatik für die Menge $\text{ART} \subseteq \Sigma^*$, die in BNF wie folgt aussah (vgl. Beispiel 6.1):

```
<digit>    ::= "0" | "1"
<bstring>  ::= "" | <digit> <bstring>
<bin>      ::= "0" | "1" <bstring>
<op2>      ::= "+" | "*" | "-"
<art>      ::= <bin> | "-" <art>
              | "(" <art> <op2> <art> ")"
```

In unserem Formalismus entspricht sie der Grammatik

$$\mathcal{G}_{\text{ART}} = (N, \Sigma_{\text{ASCII}}, P, A)$$

mit

► $N = \{A, B, C, D, O\},$

Beispiel 6.3

Wir betrachten noch einmal die Grammatik für die Menge $\text{ART} \subseteq \Sigma^*$, die in BNF wie folgt aussah (vgl. Beispiel 6.1):

```
<digit>    ::= "0" | "1"
<bstring>  ::= "" | <digit> <bstring>
<bin>      ::= "0" | "1" <bstring>
<op2>      ::= "+" | "*" | "-"
<art>      ::= <bin> | "-" <art>
              | "(" <art> <op2> <art> ")"
```

In unserem Formalismus entspricht sie der Grammatik

$$\mathcal{G}_{\text{ART}} = (N, \Sigma_{\text{ASCII}}, P, A)$$

mit

- ▶ $N = \{A, B, C, D, O\}$,
- ▶ $P = \left\{ \begin{array}{llll} D \rightarrow 0, & D \rightarrow 1, & C \rightarrow \varepsilon, & C \rightarrow DC, \\ B \rightarrow 0, & B \rightarrow 1C, & O \rightarrow +, & O \rightarrow *, \\ O \rightarrow -, & A \rightarrow B, & A \rightarrow -A, & A \rightarrow (AOA) \end{array} \right\}$

- ▶ Grammatiken bezeichnen wir mit \mathcal{G} und Varianten wie \mathcal{G}' .

- ▶ Grammatiken bezeichnen wir mit \mathcal{G} und Varianten wie \mathcal{G}' .
- ▶ Großbuchstaben A, B, \dots , die wir zur besseren Lesbarkeit farbig darstellen, bezeichnen Nichtterminalsymbole.

- ▶ Grammatiken bezeichnen wir mit \mathcal{G} und Varianten wie \mathcal{G}' .
- ▶ Großbuchstaben A, B, \dots , die wir zur besseren Lesbarkeit farbig darstellen, bezeichnen Nichtterminalsymbole.
- ▶ Kleinbuchstaben a, b, \dots bezeichnen Terminalsymbole.

- ▶ Grammatiken bezeichnen wir mit \mathcal{G} und Varianten wie \mathcal{G}' .
- ▶ Großbuchstaben A, B, \dots , die wir zur besseren Lesbarkeit farbig darstellen, bezeichnen Nichtterminalsymbole.
- ▶ Kleinbuchstaben a, b, \dots bezeichnen Terminalsymbole.
- ▶ Kleinbuchstaben u, v, \dots bezeichnen Wörter in Σ^* .
Wir bezeichnen solche Wörter als **Terminalwörter**.

- ▶ Grammatiken bezeichnen wir mit \mathcal{G} und Varianten wie \mathcal{G}' .
- ▶ Großbuchstaben A, B, \dots , die wir zur besseren Lesbarkeit farbig darstellen, bezeichnen Nichtterminalsymbole.
- ▶ Kleinbuchstaben a, b, \dots bezeichnen Terminalsymbole.
- ▶ Kleinbuchstaben u, v, \dots bezeichnen Wörter in Σ^* .
Wir bezeichnen solche Wörter als **Terminalwörter**.
- ▶ Griechische Buchstaben α, β, \dots bezeichnen Wörter in $(N \cup \Sigma)^*$.
Wir bezeichnen solche Wörter als **Satzformen**.

Kurznotation einer Grammatik

Wenn Terminal- und Nichtterminalsymbole klar sind, genügt die Auflistung der Regeln

Kurznotation einer Grammatik

Wenn Terminal- und Nichtterminalsymbole klar sind, genügt die Auflistung der Regeln

- ▶ beginnend mit einer Regel, die links das Startsymbol hat,

Kurznotation einer Grammatik

Wenn Terminal- und Nichtterminalsymbole klar sind, genügt die Auflistung der Regeln

- ▶ beginnend mit einer Regel, die links das Startsymbol hat,
- ▶ zusammengefasst nach gleichen linken Seiten, dann rechte Seiten durch „|“ getrennt

Kurznotation einer Grammatik

Wenn Terminal- und Nichtterminalsymbole klar sind, genügt die Auflistung der Regeln

- ▶ beginnend mit einer Regel, die links das Startsymbol hat,
- ▶ zusammengefasst nach gleichen linken Seiten, dann rechte Seiten durch „|“ getrennt

Beispiel 6.3 (Forts.)

Die Grammatik \mathcal{G}_{ART} lässt sich damit wie folgt darstellen:

$$A \rightarrow B \mid -A \mid (AOA)$$

$$O \rightarrow + \mid * \mid -$$

$$B \rightarrow 0 \mid 1C$$

$$C \rightarrow \varepsilon \mid DC$$

$$D \rightarrow 0 \mid 1$$

Definition 6.4

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik, und seien $\alpha, \beta \in (N \cup \Sigma)^*$ Satzformen.

Definition 6.4

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik, und seien $\alpha, \beta \in (N \cup \Sigma)^*$ Satzformen.

1. β ist **direkt herleitbar aus** α (kurz: $\alpha \rightarrow_{\mathcal{G}} \beta$), wenn es eine Regel $A \rightarrow \delta$ in P und Satzformen $\gamma_1, \gamma_2 \in (N \cup \Sigma)^*$ gibt, so dass

$$\alpha = \gamma_1 A \gamma_2 \quad \text{und} \quad \beta = \gamma_1 \delta \gamma_2.$$

Intuitiv entsteht also β aus α , indem man ein A durch δ ersetzt.

Definition 6.4

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik, und seien $\alpha, \beta \in (N \cup \Sigma)^*$ Satzformen.

1. β ist **direkt herleitbar aus** α (kurz: $\alpha \rightarrow_{\mathcal{G}} \beta$), wenn es eine Regel $A \rightarrow \delta$ in P und Satzformen $\gamma_1, \gamma_2 \in (N \cup \Sigma)^*$ gibt, so dass

$$\alpha = \gamma_1 A \gamma_2 \quad \text{und} \quad \beta = \gamma_1 \delta \gamma_2.$$

Intuitiv entsteht also β aus α , indem man ein A durch δ ersetzt.

2. Eine **Ableitung von β aus α** ist eine Folge $(\alpha_0, \dots, \alpha_n)$ von Satzformen, so dass $\alpha_0 = \alpha$, $\alpha_n = \beta$ und

$$\alpha_{i-1} \rightarrow_{\mathcal{G}} \alpha_i \quad \text{für } 1 \leq i \leq n.$$

Definition 6.4

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik, und seien $\alpha, \beta \in (N \cup \Sigma)^*$ Satzformen.

1. β ist **direkt herleitbar aus** α (kurz: $\alpha \rightarrow_{\mathcal{G}} \beta$), wenn es eine Regel $A \rightarrow \delta$ in P und Satzformen $\gamma_1, \gamma_2 \in (N \cup \Sigma)^*$ gibt, so dass

$$\alpha = \gamma_1 A \gamma_2 \quad \text{und} \quad \beta = \gamma_1 \delta \gamma_2.$$

Intuitiv entsteht also β aus α , indem man ein A durch δ ersetzt.

2. Eine **Ableitung von β aus α** ist eine Folge $(\alpha_0, \dots, \alpha_n)$ von Satzformen, so dass $\alpha_0 = \alpha$, $\alpha_n = \beta$ und

$$\alpha_{i-1} \rightarrow_{\mathcal{G}} \alpha_i \quad \text{für } 1 \leq i \leq n.$$

3. Eine **Ableitung von β** ist eine Ableitung von β aus S .

Definition 6.4

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik, und seien $\alpha, \beta \in (N \cup \Sigma)^*$ Satzformen.

1. β ist **direkt herleitbar aus** α (kurz: $\alpha \rightarrow_{\mathcal{G}} \beta$), wenn es eine Regel $A \rightarrow \delta$ in P und Satzformen $\gamma_1, \gamma_2 \in (N \cup \Sigma)^*$ gibt, so dass

$$\alpha = \gamma_1 A \gamma_2 \quad \text{und} \quad \beta = \gamma_1 \delta \gamma_2.$$

Intuitiv entsteht also β aus α , indem man ein A durch δ ersetzt.

2. Eine **Ableitung von β aus α** ist eine Folge $(\alpha_0, \dots, \alpha_n)$ von Satzformen, so dass $\alpha_0 = \alpha$, $\alpha_n = \beta$ und

$$\alpha_{i-1} \rightarrow_{\mathcal{G}} \alpha_i \quad \text{für } 1 \leq i \leq n.$$

3. Eine **Ableitung von β** ist eine Ableitung von β aus S .
4. β ist **ableitbar** aus α in \mathcal{G} (wir schreiben $\alpha \xrightarrow{*}_{\mathcal{G}} \beta$), wenn es eine Ableitung von β aus α gibt.

Definition 6.4

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik, und seien $\alpha, \beta \in (N \cup \Sigma)^*$ Satzformen.

1. β ist **direkt herleitbar aus** α (kurz: $\alpha \rightarrow_{\mathcal{G}} \beta$), wenn es eine Regel $A \rightarrow \delta$ in P und Satzformen $\gamma_1, \gamma_2 \in (N \cup \Sigma)^*$ gibt, so dass

$$\alpha = \gamma_1 A \gamma_2 \quad \text{und} \quad \beta = \gamma_1 \delta \gamma_2.$$

Intuitiv entsteht also β aus α , indem man ein A durch δ ersetzt.

2. Eine **Ableitung von β aus α** ist eine Folge $(\alpha_0, \dots, \alpha_n)$ von Satzformen, so dass $\alpha_0 = \alpha$, $\alpha_n = \beta$ und

$$\alpha_{i-1} \rightarrow_{\mathcal{G}} \alpha_i \quad \text{für } 1 \leq i \leq n.$$

3. Eine **Ableitung von β** ist eine Ableitung von β aus S .
4. β ist **ableitbar** aus α in \mathcal{G} (wir schreiben $\alpha \xrightarrow{*}_{\mathcal{G}} \beta$), wenn es eine Ableitung von β aus α gibt.
5. β ist **ableitbar** in \mathcal{G} , wenn es eine Ableitung von β gibt.

Kontextfreie Grammatiken (Semantik)

Definition 6.5

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik.

Die von \mathcal{G} erzeugte Sprache ist

$$L(\mathcal{G}) = \{w \in \Sigma^* \mid S \xrightarrow{*}_{\mathcal{G}} w\},$$

also die Menge aller in \mathcal{G} ableitbaren Terminalwörter.

Kontextfreie Grammatiken (Semantik)

Definition 6.5

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik.

Die von \mathcal{G} erzeugte Sprache ist

$$L(\mathcal{G}) = \{w \in \Sigma^* \mid S \xrightarrow{*}_{\mathcal{G}} w\},$$

also die Menge aller in \mathcal{G} ableitbaren Terminalwörter.

Terminologie und Notation

Sei \mathcal{G} eine kontextfreie Grammatik.

- Die Länge einer Ableitung $(\alpha_0, \dots, \alpha_n)$ in \mathcal{G} ist n .

Wir schreiben $\alpha \xrightarrow{n}_{\mathcal{G}} \beta$, wenn es eine Ableitung der Länge n von β aus α gibt und $\alpha \xrightarrow{\leq n}_{\mathcal{G}} \beta$, wenn es eine Ableitung der Länge höchstens n von β aus α gibt.

Kontextfreie Grammatiken (Semantik)

Definition 6.5

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik.

Die von \mathcal{G} erzeugte Sprache ist

$$L(\mathcal{G}) = \{w \in \Sigma^* \mid S \xrightarrow{*}_{\mathcal{G}} w\},$$

also die Menge aller in \mathcal{G} ableitbaren Terminalwörter.

Terminologie und Notation

Sei \mathcal{G} eine kontextfreie Grammatik.

- ▶ Die Länge einer Ableitung $(\alpha_0, \dots, \alpha_n)$ in \mathcal{G} ist n .
Wir schreiben $\alpha \xrightarrow{n}_{\mathcal{G}} \beta$, wenn es eine Ableitung der Länge n von β aus α gibt und $\alpha \xrightarrow{\leq n}_{\mathcal{G}} \beta$, wenn es eine Ableitung der Länge höchstens n von β aus α gibt.
- ▶ In den Notationen $\rightarrow_{\mathcal{G}}$, $\xrightarrow{n}_{\mathcal{G}}$, $\xrightarrow{\leq n}_{\mathcal{G}}$ und $\xrightarrow{*}_{\mathcal{G}}$ lassen wir den Index \mathcal{G} weg, wenn die Grammatik \mathcal{G} aus dem Kontext hervorgeht.

Definition 6.6

Eine Sprache ist **kontextfrei**, wenn sie von einer kontextfreien Grammatik erzeugt wird.

Definition 6.6

Eine Sprache ist **kontextfrei**, wenn sie von einer kontextfreien Grammatik erzeugt wird.

Beispiel 6.7

Sei $L_1 = \{a^n b^n \mid n \geq 0\}$.

L_1 wird von folgender Grammatik \mathcal{G}_1 erzeugt und ist damit kontextfrei:

$$S \rightarrow aSb, \quad S \rightarrow \varepsilon.$$

Definition 6.6

Eine Sprache ist **kontextfrei**, wenn sie von einer kontextfreien Grammatik erzeugt wird.

Beispiel 6.7

Sei $L_1 = \{a^n b^n \mid n \geq 0\}$.

L_1 wird von folgender Grammatik \mathcal{G}_1 erzeugt und ist damit kontextfrei:

$$S \rightarrow aSb, \quad S \rightarrow \varepsilon.$$

Die Ableitungen einiger Wörter in \mathcal{G}_1 sind:

$$\begin{aligned} S &\rightarrow \varepsilon, & S &\rightarrow aSb \rightarrow a\varepsilon b = ab, \\ S &\rightarrow aSb \rightarrow aaSbb \rightarrow aaaSbbb \rightarrow aaabbbb. \end{aligned}$$

Definition 6.6

Eine Sprache ist **kontextfrei**, wenn sie von einer kontextfreien Grammatik erzeugt wird.

Beispiel 6.7

Sei $L_1 = \{a^n b^n \mid n \geq 0\}$.

L_1 wird von folgender Grammatik \mathcal{G}_1 erzeugt und ist damit kontextfrei:

$$S \rightarrow aSb, \quad S \rightarrow \varepsilon.$$

Die Ableitungen einiger Wörter in \mathcal{G}_1 sind:

$$\begin{aligned} S &\rightarrow \varepsilon, & S &\rightarrow aSb \rightarrow a\varepsilon b = ab, \\ S &\rightarrow aSb \rightarrow aaSbb \rightarrow aaaSbbb \rightarrow aaabbb. \end{aligned}$$

Mit Hilfe des Pumpinglemmas lässt sich leicht zeigen, dass L_1 nicht regulär ist.

Beispiel 6.3 (Forts.)

Die Sprache ART der arithmetischen Terme ist kontextfrei. Sie wird erzeugt von der Grammatik

$$\begin{aligned}A &\rightarrow B \mid -A \mid (AOA) \\B &\rightarrow 0 \mid 1C \\D &\rightarrow 0 \mid 1\end{aligned}$$

$$\begin{aligned}O &\rightarrow + \mid * \mid - \\C &\rightarrow \varepsilon \mid DC\end{aligned}$$

Beispiel 6.3 (Forts.)

Die Sprache ART der arithmetischen Terme ist kontextfrei. Sie wird erzeugt von der Grammatik

$$\begin{aligned}A &\rightarrow B \mid -A \mid (AOA) \\B &\rightarrow 0 \mid 1C \\D &\rightarrow 0 \mid 1\end{aligned}$$

$$\begin{aligned}O &\rightarrow + \mid * \mid - \\C &\rightarrow \varepsilon \mid DC\end{aligned}$$

Eine Ableitung des Terms $-(10*(0+1))$ ist

$$\begin{aligned}A &\rightarrow -A \rightarrow -(AOA) \rightarrow -(BOA) \rightarrow -(1COA) \rightarrow -(1DCOA) \\&\rightarrow -(10COA) \rightarrow -(10QA) \rightarrow -(10*A) \rightarrow -(10*(AOA)) \\&\rightarrow -(10*(BOA)) \rightarrow -(10*(0QA)) \rightarrow -(10*(0+A)) \\&\rightarrow -(10*(0+B)) \rightarrow -(10*(0+1C)) \rightarrow -(10*(0+1))\end{aligned}$$

Hier ist jeweils das Nichtterminalsymbol, das als nächstes ersetzt wird, unterstrichen.

Ein **Palindrom** ist ein Wort, das von vorne und hinten gelesen gleich ist.

Ein **Palindrom** ist ein Wort, das von vorne und hinten gelesen gleich ist.

Wir betrachten die Sprache aller Palindrome über dem Alphabet $\{a, b\}$:

$$L_P := \{a_1 \dots a_n \in \{a, b\}^* \mid i \in \mathbb{N}, a_i = a_{n+1-i} \text{ für } 1 \leq i \leq n\}.$$

Ein **Palindrom** ist ein Wort, das von vorne und hinten gelesen gleich ist.

Wir betrachten die Sprache aller Palindrome über dem Alphabet $\{a, b\}$:

$$L_P := \{a_1 \dots a_n \in \{a, b\}^* \mid i \in \mathbb{N}, a_i = a_{n+1-i} \text{ für } 1 \leq i \leq n\}.$$

L_P wird von folgender kontextfreier Grammatik \mathcal{G}_P erzeugt und ist damit kontextfrei:

$$S \rightarrow aSa \mid bSb \mid a \mid b \mid \varepsilon.$$

Lemma 6.9 (Kombinationslemma)

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Seien $\alpha, \beta, \gamma, \delta \in (N \cup \Sigma)^*$ und $A \in N$, so dass

$$\alpha \xrightarrow{*} \beta A \delta \quad \text{und} \quad A \xrightarrow{*} \gamma.$$

Lemma 6.9 (Kombinationslemma)

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Seien $\alpha, \beta, \gamma, \delta \in (N \cup \Sigma)^*$ und $A \in N$, so dass

$$\alpha \xrightarrow{*} \beta A \delta \quad \text{und} \quad A \xrightarrow{*} \gamma.$$

Dann gilt $\alpha \xrightarrow{*} \beta \gamma \delta$.

Lemma 6.9 (Kombinationslemma)

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Seien $\alpha, \beta, \gamma, \delta \in (N \cup \Sigma)^*$ und $A \in N$, so dass

$$\alpha \xrightarrow{*} \beta A \delta \quad \text{und} \quad A \xrightarrow{*} \gamma.$$

Dann gilt $\alpha \xrightarrow{*} \beta \gamma \delta$.

Lemma 6.10 (Zerlegungslemma)

Sei $n \in \mathbb{N}$. Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Seien $\alpha_1, \alpha_2, \beta \in (N \cup \Sigma)^*$ Satzformen, so dass

$$\alpha_1 \alpha_2 \xrightarrow{n} \beta$$

Dann gibt es eine Zerlegung $\beta = \beta_1 \beta_2$, so dass

$$\alpha_1 \xrightarrow{\leq n} \beta_1 \quad \text{und} \quad \alpha_2 \xrightarrow{\leq n} \beta_2.$$

Beweis des Kombinationslemmas.

$$A \xrightarrow{*} \gamma$$

Sei $(\gamma_0 = A, \gamma_1, \dots, \gamma_m = \gamma)$ eine Ableitung von γ aus A . Dann ist

$$(\beta\gamma_0\delta, \beta\gamma_1\delta, \dots, \beta\gamma_m\delta)$$

eine Ableitung von $\beta\gamma\delta$ aus $\beta A\delta$.

$$\alpha \xrightarrow{*} \beta A \delta$$

Sei $(\alpha_0 = \alpha, \alpha_1, \dots, \alpha_\ell = \beta A\delta)$ eine Ableitung von $\beta A\delta$ aus α . Dann ist

$$(\alpha_0 = \alpha, \alpha_1, \dots, \alpha_\ell = \beta\gamma_0\delta, \beta\gamma_1\delta, \dots, \beta\gamma_m\delta)$$

eine Ableitung von $\beta\gamma\delta$ aus α . □

Beweis des Zerlegungslemmas.

Induktion über die Länge n der Ableitung.

Induktionsanfang: $n = 0$

$$\alpha_1 \alpha_2 \xrightarrow{n} \beta$$

Gelte $\alpha_1 \alpha_2 \xrightarrow{0} \beta$. Dann gilt $\beta = \alpha_1 \alpha_2$, und wir setzen $\beta_1 := \alpha_1$ und $\beta_2 := \alpha_2$.

Induktionsschritt: $n = n + 1$

Gelte $\alpha_1 \alpha_2 \xrightarrow{n+1} \beta$. Dann gibt es ein $\beta' \in (N \cup \Sigma)^*$, so dass

$$\alpha_1 \alpha_2 \xrightarrow{n} \beta' \rightarrow \beta$$

Nach Induktionsannahme gibt es eine Zerlegung $\beta' = \beta'_1 \beta'_2$, so dass

$$\alpha_1 \xrightarrow{\leq n} \beta'_1 \quad \text{und} \quad \alpha_2 \xrightarrow{\leq n} \beta'_2.$$

Betrachten wir den letzten Schritt der Ableitung, $\beta' \rightarrow \beta$. Dann gibt es eine Zerlegung $\beta' = \gamma_1 A \gamma_2$ und eine Regel $A \rightarrow \delta \in P$, so dass $\beta = \gamma_1 \delta \gamma_2$.

Wir unterscheiden zwei Fälle: Das A in $\gamma_1 A \gamma_2$ gehört zu β'_1 , und das A in $\gamma_1 A \gamma_2$ gehört zu β'_2 .

Fall 1: Das A in $\gamma_1 A \gamma_2$ gehört zu β'_1 .

Dann gibt es eine Zerlegung $\gamma_2 = \gamma_{21} \gamma_{22}$, so dass

$$\beta'_1 = \gamma_1 A \gamma_{21} \quad \text{und} \quad \beta'_2 = \gamma_{22}.$$

Es gilt $\beta = \gamma_1 \delta \gamma_{21} \gamma_{22}$ und $\beta'_1 \rightarrow \gamma_1 \delta \gamma_{21}$. Wir setzen

$$\beta_1 := \gamma_1 \delta \gamma_{21} \quad \text{und} \quad \beta_2 := \gamma_{22}$$

Dann gilt $\beta = \beta_1 \beta_2$ und $\alpha_1 \xrightarrow{\leq n} \beta'_1 \rightarrow \beta_1$, also $\alpha_1 \xrightarrow{\leq n+1} \beta_1$ und $\alpha_2 \xrightarrow{\leq n} \beta'_2 = \beta_2$.

Fall 2: Das A in $\gamma_1 A \gamma_2$ gehört zu β'_2 .

Dann gibt es eine Zerlegung $\gamma_1 = \gamma_{11} \gamma_{12}$, so dass

$$\beta'_1 = \gamma_{11} \quad \text{und} \quad \beta'_2 = \gamma_{12} A \gamma_2$$

Es gilt $\beta = \gamma_{11} \gamma_{12} \delta \gamma_2$ und $\beta'_2 \rightarrow \gamma_{12} \delta \gamma_2$. Wir setzen

$$\beta_1 := \gamma_{11} \quad \text{und} \quad \beta_2 := \gamma_{12} \delta \gamma_2$$

Dann gilt $\beta = \beta_1 \beta_2$ und $\alpha_1 \xrightarrow{\leq n} \beta'_1 = \beta_1$ und $\alpha_2 \xrightarrow{\leq n} \beta'_2 \rightarrow \beta_2$, also $\alpha_2 \xrightarrow{\leq n+1} \beta_2$. □

Die Sprache $L_K \subseteq \{ (,) \}$ aller korrekten Klammerausdrücke ist kontextfrei. Sie wird von folgender Grammatik \mathcal{G}_k erzeugt:

$$\begin{aligned} S &\rightarrow (A) \mid \varepsilon \\ A &\rightarrow S \mid AA \end{aligned}$$

Behauptung

$$L_K = L(\mathcal{G}_K)$$

Beweis.

$L_K \subseteq L(\mathcal{G}_k)$: Wir zeigen induktiv über den Aufbau von $w \in L_K$, dass $w \in L(\mathcal{G}_k)$.

Induktionsanfang: $S \rightarrow \varepsilon$ ist eine Ableitung von ε in \mathcal{G}_k , also gilt $\varepsilon \in L(\mathcal{G}_k)$.

Induktionsschritt: Seien $w_1, \dots, w_k \in L_K$ und $w = (w_1 \dots w_k)$. Nach Induktionsannahme gilt für $1 \leq i \leq k$

$$S \xrightarrow{*} w_i.$$

Außerdem ist $(\overbrace{S \dots S}^{k \text{ mal}})$ ableitbar:

$$\begin{aligned} S &\rightarrow (A) \rightarrow (AA) \rightarrow (AAA) \rightarrow \dots \rightarrow (AA \dots A) \\ &\rightarrow (SA \dots A) \rightarrow \dots \rightarrow (S \dots S) \end{aligned}$$

Jetzt ergibt k -maliges Anwenden des Kombinationslemmas eine Ableitung von w .

$L(\mathcal{G}_k) \subseteq L_K$: Wir zeigen per Induktion über die Länge n einer Ableitung, für alle Terminalwörter $w \in \Sigma^*$:

- (i) Wenn $S \xrightarrow{n} w$, dann $w \in L_K$.
- (ii) Wenn $A \xrightarrow{n} w$, dann $w \in L_K^*$.

Induktionsanfang: $n = 1$

- (i) Wenn $S \rightarrow w$ für ein $w \in \Sigma^*$, so $w = \varepsilon \in L_K$.
- (ii) $A \rightarrow w \in \Sigma^*$ ist nicht möglich.

Induktionsschritt: $1, \dots, n \rightarrow n + 1$.

- (i) Gelte $S \xrightarrow{n+1} w$. Dann gilt $S \rightarrow (A) \xrightarrow{n} w$. Nach dem Zerlegungslemma (zweimal angewandt) gibt es eine Zerlegung $w = xyz$, so dass

$$(\xrightarrow{\leq n} x \quad \text{und} \quad A \xrightarrow{\leq n} y \quad \text{und} \quad) \xrightarrow{\leq n} z$$

Weil $($ und $)$ keine Nichtterminale enthalten, gilt $x = ($ und $z =)$.

Nach Induktionsannahme gilt $y \in L_K^*$, also $y = y_1 \dots y_k$ für ein $k \geq 1$ und $y_1, \dots, y_k \in L_K$. Hier können wir annehmen, dass $k \geq 1$, weil $\varepsilon \in L_K$.

Insgesamt gilt also

$$w = (y_1 \dots y_k)$$

für ein $k \geq 1$ und $y_1, \dots, y_k \in L_K$.

Weil L_K abgeschlossen ist unter der rekursiven Regel

„Wenn $w_1, \dots, w_k \in L_K$, dann $(w_1 \dots w_k)$.“

gilt damit $w \in L_K$.

- (ii) Gelte $A \xrightarrow{n+1} w$. Dann gilt entweder $A \rightarrow S \xrightarrow{n} w$ oder $A \rightarrow AA \xrightarrow{n} w$.

Falls $S \xrightarrow{n} w$, so $w \in L_K$ nach Induktionsannahme.

Nehmen wir also an, dass $AA \xrightarrow{n} w$. Nach dem Zerlegungslemma gibt es eine Zerlegung $w = xy$, so dass

$$A \xrightarrow{\leq n} x \quad \text{und} \quad A \xrightarrow{\leq n} y.$$

Nach Induktionsannahme gilt dann $x \in L_K^*$ und $y \in L_K^*$ und damit $w = xy \in L_K^*$.

□

Satz 6.12

Alle regulären Sprachen sind kontextfrei.

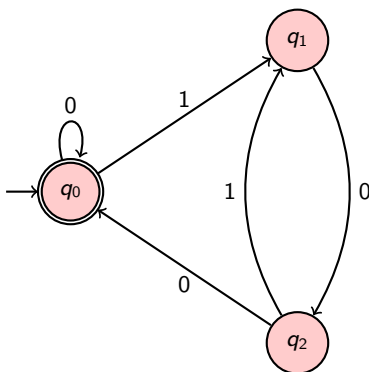
Satz 6.12

Alle regulären Sprachen sind kontextfrei.

Bemerkung 6.13

Die Umkehrung dieses Satzes gilt nicht. Wir haben bereits einige Beispiele von nichtregulären Sprachen gesehen, die kontextfrei sind.

Sei $L = L(\mathcal{A})$ für folgenden NFA \mathcal{A} .



Wir wollen eine kontextfreie Grammatik angeben, die L erzeugt.

Wir notieren dazu Läufe des NFAs durch Präfixe des gelesenen Wortes und die jeweils erreichten Zustände.

Beispiel: Für den Lauf

$$(q_0, 1, q_1, 0, q_2, 1, q_1, 0, q_2, 0, q_0)$$

schreiben wir

$$q_0 \rightarrow 1q_1 \rightarrow 10q_2 \rightarrow 101q_1 \rightarrow 1010q_2 \rightarrow 10100q_0 \rightarrow 10100 \quad (*)$$

Jetzt konstruieren wir eine Grammatik, deren Nichtterminalsymbole die Zustände des NFAs sind, so dass $(*)$ eine Ableitung dieser Grammatik ist:

$$q_0 \rightarrow 0q_0 \mid 1q_1 \mid \varepsilon$$

$$q_1 \rightarrow 0q_2$$

$$q_2 \rightarrow 0q_0 \mid 1q_1.$$

Definition 6.15

Für jeden NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ sei

$$\mathcal{G}_{\mathcal{A}} = (Q, \Sigma, P, q_0)$$

die kontextfreie Grammatik mit

$$P := \{ q \rightarrow ar \mid (q, a, r) \in \Delta \} \cup \{ q \rightarrow \varepsilon \mid q \in F \}.$$

Definition 6.15

Für jeden NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ sei

$$\mathcal{G}_{\mathcal{A}} = (Q, \Sigma, P, q_0)$$

die kontextfreie Grammatik mit

$$P := \{ q \rightarrow ar \mid (q, a, r) \in \Delta \} \cup \{ q \rightarrow \varepsilon \mid q \in F \}.$$

Lemma 6.16

Für jeden NFA \mathcal{A} gilt $L(\mathcal{A}) = L(\mathcal{G}_{\mathcal{A}})$.

Definition 6.15

Für jeden NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ sei

$$\mathcal{G}_{\mathcal{A}} = (Q, \Sigma, P, q_0)$$

die kontextfreie Grammatik mit

$$P := \{q \rightarrow ar \mid (q, a, r) \in \Delta\} \cup \{q \rightarrow \varepsilon \mid q \in F\}.$$

Lemma 6.16

Für jeden NFA \mathcal{A} gilt $L(\mathcal{A}) = L(\mathcal{G}_{\mathcal{A}})$.

Satz 6.12 folgt sofort aus dem Lemma.

$$\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$$

$\mathcal{G}_{\mathcal{A}}$:

$$\begin{array}{ll} q \rightarrow ar & \text{für alle } (q, a, r) \in \Delta, \\ q \rightarrow \varepsilon & \text{für alle } q \in F. \end{array}$$

Behauptung

$$L(\mathcal{A}) = L(\mathcal{G}_{\mathcal{A}})$$

Beweis.

„ \subseteq “ Für jeden akzeptierenden Lauf

$$(q_0, a_1, q_1, \dots, a_n, q_n)$$

auf einem Wort $w = a_1 \dots a_n \in L(\mathcal{A})$ ist

$$q_0 \rightarrow a_1 q_1 \rightarrow \dots \rightarrow a_1 \dots a_i q_i \rightarrow \dots \rightarrow a_1 \dots a_n q_n \rightarrow a_1 \dots a_n$$

eine Ableitung von w in $\mathcal{G}_{\mathcal{A}}$. Also $w \in L(\mathcal{G}_{\mathcal{A}})$.

„ \supseteq “ Wir beweisen per Induktion über $n \geq 0$, dass für jede Satzform $\alpha \in (Q \cup \Sigma)^*$ mit

$$q_0 \xrightarrow{n} \alpha$$

gilt:

(i) entweder $\alpha = wq$ für ein $w \in \Sigma^*$ und ein $q \in Q$, so dass $\mathcal{A} : q_0 \xrightarrow{w} q$,

(ii) oder $\alpha = w$ für ein $w \in L(\mathcal{A})$.

Für alle ableitbaren Terminalwörter $w \in \Sigma^*$ sind wir also im Fall (ii), und es gilt $w \in L(\mathcal{A})$. Also $L(\mathcal{G}_{\mathcal{A}}) \subseteq L(\mathcal{A})$.

Induktionsanfang: $n = 0$.

$q_0 \xrightarrow{0} \alpha$ impliziert $\alpha = q_0$, und weil $\mathcal{A} : q_0 \xrightarrow{\varepsilon} q_0$ gilt (i).

Induktionsschritt: $n \rightarrow n + 1$.

Gelte $q_0 \xrightarrow{n+1} \alpha$. Dann gibt es ein $\alpha' \in (Q \cup \Sigma)^*$ mit

$$q_0 \xrightarrow{n} \alpha' \rightarrow \alpha$$

Nach Induktionsannahme erfüllt α' (i) oder (ii), und weil $\alpha' \rightarrow \alpha$ muss α' mindestens ein Nichtterminalsymbol enthalten, erfüllt also (i). Das heißt,

$$\alpha' = wq$$

für ein $w \in \Sigma^*$ und ein $q \in Q$, so dass $\mathcal{A} : q_0 \xrightarrow{w} q$. Die Regel im letzten Schritt $\alpha' \rightarrow \alpha$ der Ableitung ist

- ▶ entweder $q \rightarrow ar$ für ein Tripel $(q, a, r) \in \Delta$
- ▶ oder $q \rightarrow \varepsilon$, falls $q \in F$.

Im ersten Fall ist $\alpha = war$, und es gilt

$$\mathcal{A} : q_0 \xrightarrow{w} q \xrightarrow{a} r.$$

Also ist (i) erfüllt.

Im zweiten Fall ist $\alpha = w$ und $w \in L(\mathcal{A})$, weil $\mathcal{A} : q_0 \xrightarrow{w} q \in F$.

Also ist (ii) erfüllt. □

Abschnitt 6.2

Normalformen

In vielen Anwendungen ist es hilfreich, nur Grammatiken in einer speziellen Form (einer **Normalform**) zu betrachten.

In vielen Anwendungen ist es hilfreich, nur Grammatiken in einer speziellen Form (einer **Normalform**) zu betrachten.

Chomsky-Normalform: Alle Regeln haben die Form

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

In vielen Anwendungen ist es hilfreich, nur Grammatiken in einer speziellen Form (einer **Normalform**) zu betrachten.

Chomsky-Normalform: Alle Regeln haben die Form

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

Greibach-Normalform: Alle Regeln haben die Form

$$A \rightarrow aB_1 \dots B_k \quad \text{für ein } k \geq 0.$$

In vielen Anwendungen ist es hilfreich, nur Grammatiken in einer speziellen Form (einer **Normalform**) zu betrachten.

Chomsky-Normalform: Alle Regeln haben die Form

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

Greibach-Normalform: Alle Regeln haben die Form

$$A \rightarrow aB_1 \dots B_k \quad \text{für ein } k \geq 0.$$

Rechtslinear: Alle Regeln haben die Form

$$A \rightarrow aB \quad \text{oder} \quad A \rightarrow a.$$

In vielen Anwendungen ist es hilfreich, nur Grammatiken in einer speziellen Form (einer **Normalform**) zu betrachten.

Chomsky-Normalform: Alle Regeln haben die Form

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

Greibach-Normalform: Alle Regeln haben die Form

$$A \rightarrow aB_1 \dots B_k \quad \text{für ein } k \geq 0.$$

Rechtslinear: Alle Regeln haben die Form

$$A \rightarrow aB \quad \text{oder} \quad A \rightarrow a.$$

In diesem Kapitel wollen wir untersuchen, inwieweit sich beliebige kontextfreie Grammatiken in äquivalente Grammatiken dieser Formen umwandeln lassen.

Um das leere Wort zu erzeugen, muss eine Grammatik mindestens eine Regel $A \rightarrow \varepsilon$ (eine ε -Regel) enthalten.

Um das leere Wort zu erzeugen, muss eine Grammatik mindestens eine Regel $A \rightarrow \varepsilon$ (eine ε -Regel) enthalten.

Das bedeutet, dass keine Grammatik in einer der auf der letzten Seite aufgeführten Formen das leere Wort erzeugen kann.

Um das leere Wort zu erzeugen, muss eine Grammatik mindestens eine Regel $A \rightarrow \varepsilon$ (eine ε -Regel) enthalten.

Das bedeutet, dass keine Grammatik in einer der auf der letzten Seite aufgeführten Formen das leere Wort erzeugen kann.

Der Einfachheit halber berücksichtigen wir in diesem Kapitel das leere Wort in den Sprachen nicht.

Um das leere Wort zu erzeugen, muss eine Grammatik mindestens eine Regel $A \rightarrow \varepsilon$ (eine ε -Regel) enthalten.

Das bedeutet, dass keine Grammatik in einer der auf der letzten Seite aufgeführten Formen das leere Wort erzeugen kann.

Der Einfachheit halber berücksichtigen wir in diesem Kapitel das leere Wort in den Sprachen nicht.

Aber:

Jede Grammatik $G = (N, \Sigma, R, S)$ lässt sich durch

- ▶ Hinzunahme eines neuen Startsymbols S' ,
- ▶ Hinzunahme der Regeln $S' \rightarrow \varepsilon \mid S$

in eine Grammatik $G' = (N, \Sigma, P', S')$ mit $L(G') = L(G) \cup \{\varepsilon\}$ transformieren.

Lemma 6.17

Zu jeder Grammatik \mathcal{G} gibt es eine Grammatik \mathcal{G}' ohne ε -Regeln, so dass $L(\mathcal{G}') = L(\mathcal{G}) \setminus \{\varepsilon\}$.

Lemma 6.17

Zu jeder Grammatik \mathcal{G} gibt es eine Grammatik \mathcal{G}' ohne ε -Regeln, so dass $L(\mathcal{G}') = L(\mathcal{G}) \setminus \{\varepsilon\}$.

Beweisskizze.

Sei $\mathcal{G} = (N, \Sigma, P, S)$.

Lemma 6.17

Zu jeder Grammatik \mathcal{G} gibt es eine Grammatik \mathcal{G}' ohne ε -Regeln, so dass $L(\mathcal{G}') = L(\mathcal{G}) \setminus \{\varepsilon\}$.

Beweisskizze.

Sei $\mathcal{G} = (N, \Sigma, P, S)$.

Sei P' die Menge von Regeln, die aus P durch wiederholtes Anwenden des folgenden Erweiterungsschrittes entsteht (bis keine Erweiterung mehr möglich ist).

Erweiterungsschritt: Sind $A \rightarrow \varepsilon$ und $B \rightarrow \alpha A \beta$ Regeln, so füge die Regel $B \rightarrow \alpha \beta$ hinzu.

Lemma 6.17

Zu jeder Grammatik \mathcal{G} gibt es eine Grammatik \mathcal{G}' ohne ε -Regeln, so dass $L(\mathcal{G}') = L(\mathcal{G}) \setminus \{\varepsilon\}$.

Beweisskizze.

Sei $\mathcal{G} = (N, \Sigma, P, S)$.

Sei P' die Menge von Regeln, die aus P durch wiederholtes Anwenden des folgenden Erweiterungsschrittes entsteht (bis keine Erweiterung mehr möglich ist).

Erweiterungsschritt: Sind $A \rightarrow \varepsilon$ und $B \rightarrow \alpha A \beta$ Regeln, so füge die Regel $B \rightarrow \alpha \beta$ hinzu.

Anschließend entfernen wir alle ε -Regeln. Sei P' die resultierende Regelmenge und $\mathcal{G}' = (N, \Sigma, P', S)$.

Lemma 6.17

Zu jeder Grammatik \mathcal{G} gibt es eine Grammatik \mathcal{G}' ohne ε -Regeln, so dass $L(\mathcal{G}') = L(\mathcal{G}) \setminus \{\varepsilon\}$.

Beweisskizze.

Sei $\mathcal{G} = (N, \Sigma, P, S)$.

Sei P' die Menge von Regeln, die aus P durch wiederholtes Anwenden des folgenden Erweiterungsschrittes entsteht (bis keine Erweiterung mehr möglich ist).

Erweiterungsschritt: Sind $A \rightarrow \varepsilon$ und $B \rightarrow \alpha A \beta$ Regeln, so füge die Regel $B \rightarrow \alpha \beta$ hinzu.

Anschließend entfernen wir alle ε -Regeln. Sei P' die resultierende Regelmenge und $\mathcal{G}' = (N, \Sigma, P', S)$.

Es ist nicht schwer (wenn auch ein wenig mühsam), zu zeigen, dass $L(\mathcal{G}') = L(\mathcal{G}) \setminus \{\varepsilon\}$. □

Beispiel 6.18

Wir betrachten folgende Grammatik \mathcal{G}_K (vgl. Beispiel 6.11), die die Sprache L_K aller korrekten Klammerausdrücke erzeugt.

$$\begin{aligned} S &\rightarrow (A) \mid \varepsilon \\ A &\rightarrow S \mid AA \end{aligned}$$

Beispiel 6.18

Wir betrachten folgende Grammatik \mathcal{G}_K (vgl. Beispiel 6.11), die die Sprache L_K aller korrekten Klammerausdrücke erzeugt.

$$\begin{aligned} S &\rightarrow (A) \mid \varepsilon \\ A &\rightarrow S \mid AA \end{aligned}$$

Anwenden der Konstruktion zur Elimination der ε -Regeln ergibt:

Erster Eliminationsschritt (mit $S \rightarrow \varepsilon$)

Füge die Regel $A \rightarrow \varepsilon$ hinzu.

Beispiel 6.18

Wir betrachten folgende Grammatik \mathcal{G}_K (vgl. Beispiel 6.11), die die Sprache L_K aller korrekten Klammerausdrücke erzeugt.

$$\begin{aligned} S &\rightarrow (A) \mid \varepsilon \\ A &\rightarrow S \mid AA \end{aligned}$$

Anwenden der Konstruktion zur Elimination der ε -Regeln ergibt:

Erster Eliminationsschritt (mit $S \rightarrow \varepsilon$)

Füge die Regel $A \rightarrow \varepsilon$ hinzu.

Zweiter Eliminationsschritt (mit $A \rightarrow \varepsilon$)

Füge die Regeln $S \rightarrow ()$ und $A \rightarrow A$ hinzu.

Beispiel 6.18

Wir betrachten folgende Grammatik \mathcal{G}_K (vgl. Beispiel 6.11), die die Sprache L_K aller korrekten Klammerausdrücke erzeugt.

$$\begin{aligned} S &\rightarrow (A) \mid \varepsilon \\ A &\rightarrow S \mid AA \end{aligned}$$

Anwenden der Konstruktion zur Elimination der ε -Regeln ergibt:

Erster Eliminationsschritt (mit $S \rightarrow \varepsilon$)

Füge die Regel $A \rightarrow \varepsilon$ hinzu.

Zweiter Eliminationsschritt (mit $A \rightarrow \varepsilon$)

Füge die Regeln $S \rightarrow ()$ und $A \rightarrow A$ hinzu.

Entferne die Regeln $S \rightarrow \varepsilon$ und $A \rightarrow \varepsilon$.

Beispiel 6.18

Wir betrachten folgende Grammatik \mathcal{G}_K (vgl. Beispiel 6.11), die die Sprache L_K aller korrekten Klammerausdrücke erzeugt.

$$\begin{aligned} S &\rightarrow (A) \mid \varepsilon \\ A &\rightarrow S \mid AA \end{aligned}$$

Anwenden der Konstruktion zur Elimination der ε -Regeln ergibt:

Erster Eliminationsschritt (mit $S \rightarrow \varepsilon$)

Füge die Regel $A \rightarrow \varepsilon$ hinzu.

Zweiter Eliminationsschritt (mit $A \rightarrow \varepsilon$)

Füge die Regeln $S \rightarrow ()$ und $A \rightarrow A$ hinzu.

Entferne die Regeln $S \rightarrow \varepsilon$ und $A \rightarrow \varepsilon$.

Es ergibt sich folgende Grammatik \mathcal{G}'_K :

$$\begin{aligned} S &\rightarrow (A) \mid () \\ A &\rightarrow S \mid AA \mid A \end{aligned}$$

Definition 6.19

Eine kontextfreie Grammatik ist in **Chomsky-Normalform (CNF)**, wenn sie nur Regeln der Form

$$A \rightarrow BC$$

$$A \rightarrow a,$$

für Nichtterminalsymbole A, B, C und Terminalsymbole a , hat.

Definition 6.19

Eine kontextfreie Grammatik ist in **Chomsky-Normalform (CNF)**, wenn sie nur Regeln der Form

$$A \rightarrow BC$$

$$A \rightarrow a,$$

für Nichtterminalsymbole A, B, C und Terminalsymbole a , hat.

Satz 6.20

Jede kontextfreie Grammatik ist äquivalent zu einer kontextfreien Grammatik in CNF.

Definition 6.19

Eine kontextfreie Grammatik ist in **Chomsky-Normalform (CNF)**, wenn sie nur Regeln der Form

$$A \rightarrow BC$$

$$A \rightarrow a,$$

für Nichtterminalsymbole A, B, C und Terminalsymbole a , hat.

Satz 6.20

Jede kontextfreie Grammatik ist äquivalent zu einer kontextfreien Grammatik in CNF.

Genauer: Zu jeder kontextfreien Grammatik \mathcal{G} gibt es eine kontextfreie Grammatik \mathcal{G}' in CNF, so dass $L(\mathcal{G}') = L(\mathcal{G}) \setminus \{\varepsilon\}$.

Vier Schritte

1. Elimination der ε -Regeln
(bereits durchgeführt, Lemma 6.17)

Vier Schritte

1. Elimination der ε -Regeln
(bereits durchgeführt, Lemma 6.17)
2. Terminalsymbole nur in Regeln der Form $A \rightarrow a$
(dann nur noch weitere Regeln $A \rightarrow A_1 \dots A_m$)

Vier Schritte

1. Elimination der ε -Regeln
(bereits durchgeführt, Lemma 6.17)
2. Terminalsymbole nur in Regeln der Form $A \rightarrow a$
(dann nur noch weitere Regeln $A \rightarrow A_1 \dots A_m$)
3. Elimination der Regeln $A \rightarrow B$

Vier Schritte

1. Elimination der ε -Regeln
(bereits durchgeführt, Lemma 6.17)
2. Terminalsymbole nur in Regeln der Form $A \rightarrow a$
(dann nur noch weitere Regeln $A \rightarrow A_1 \dots A_m$)
3. Elimination der Regeln $A \rightarrow B$
4. Elimination der Regeln $A \rightarrow A_1 \dots A_m$ mit $m > 2$

Als Ergebnis von Schritt 1 haben wir eine kontextfreie Grammatik \mathcal{G}_1 ohne ε -Regeln.

Ziel

Konstruktion einer zu \mathcal{G}_1 äquivalenten Grammatik \mathcal{G}_2 , die nur Regeln der Formen

$$A \rightarrow a \quad \text{und} \quad A \rightarrow A_1 \dots A_m$$

mit $A, A_1, \dots, A_m \in N$, $m \geq 1$ und $a \in \Sigma$ enthält

Beispiel 6.21

Aus der Grammatik \mathcal{G}_K für die Sprache L_K der korrekten Klammerausdrücke erhalten wir in Schritt 1 durch Elimination der ε -Transitionen folgende Grammatik \mathcal{G}_1 :

$$\begin{aligned} S &\rightarrow (A) \mid () \\ A &\rightarrow S \mid AA \mid A \end{aligned}$$

(siehe Beispiel 6.18).

Für die beiden Terminalsymbole $(,)$ führen wir zwei neue Nichtterminalsymbole $T_()$ und $T_)$ ein und verwenden diese an Stelle von $($ (bzw. $)$).

Außerdem fügen wir Regeln $T_() \rightarrow ($ und $T_() \rightarrow)$ hinzu.

Es ergibt sich folgende Grammatik \mathcal{G}_2 :

$$\begin{aligned} S &\rightarrow T_()AT_() \mid T_()T_() \\ A &\rightarrow S \mid AA \mid A \\ T_() &\rightarrow (\\ T_() &\rightarrow) \end{aligned}$$

Allgemeine Konstruktion

Wir bilden die Grammatik \mathcal{G}_2 aus \mathcal{G}_1 wie folgt:

- ▶ Für jedes Terminalsymbol $a \in \Sigma$ führen wir ein neues Nichtterminalsymbol T_a ein.
- ▶ In allen Regeln von \mathcal{G}_1 ersetzen wir überall a durch T_a , für alle Terminalsymbole $a \in \Sigma$.
- ▶ Für alle Terminalsymbole $a \in \Sigma$ fügen wir die Regel $T_a \rightarrow a$ hinzu.

Dann lässt sich leicht beweisen, dass $L(\mathcal{G}_1) = L(\mathcal{G}_2)$.

Als Ergebnis von Schritt 2 haben wir eine kontextfreie Grammatik

$$\mathcal{G}_2 = (N_2, \Sigma, P_2, S_2)$$

die nur Regeln der Formen

$$A \rightarrow a \quad \text{und} \quad A \rightarrow A_1 \dots A_m$$

mit $A, A_1, \dots, A_m \in N$, $m \geq 1$ und $a \in \Sigma$ enthält.

Ziel

Konstruktion einer zu \mathcal{G}_2 äquivalenten Grammatik \mathcal{G}_3 , die nur Regeln der Formen

$$A \rightarrow a \quad \text{und} \quad A \rightarrow A_1 \dots A_m$$

mit $A, A_1, \dots, A_m \in N$, $m \geq 2$ und $a \in \Sigma$ enthält

Konstruktion

Wir bilden die Grammatik \mathcal{G}_3 aus \mathcal{G}_2 wie folgt:

- Für jede Ableitung

$$A_1 \rightarrow_{\mathcal{G}_2} A_2 \rightarrow_{\mathcal{G}_2} \dots \rightarrow_{\mathcal{G}_2} A_n \rightarrow_{\mathcal{G}_2} \alpha$$

mit $n \geq 2$ und $A_1, \dots, A_n \in N_2$ und $\alpha \notin N_2$ fügen wir die Regel $A_1 \rightarrow \alpha$ hinzu.

- Wir entfernen alle Regeln der Form $A \rightarrow B$, für $A, B \in N_2$.

Dann lässt sich leicht beweisen, dass $L(\mathcal{G}_2) = L(\mathcal{G}_3)$.

Praktische Umsetzung

Wie finden wir alle Ableitungen $A_1 \rightarrow_{\mathcal{G}_2} A_2 \rightarrow_{\mathcal{G}_2} \dots \rightarrow_{\mathcal{G}_2} A_n \rightarrow_{\mathcal{G}_2} \alpha$ mit $n \geq 2$ und $A_1, \dots, A_n \in N_2$ und $\alpha \notin N_2$?

Und sind es nicht viel zu viele (es kann ja sogar unendlich viele solcher Ableitungen geben)?

Wir konstruieren einen gerichteten Graphen $D = (V, E)$ wie folgt:

- Die Knotenmenge V besteht aus allen Nichtterminalen von \mathcal{G}_2 und allen Satzformen $\alpha \notin N_2$, die auf der rechten Seite einer Regel von \mathcal{G}_2 auftauchen.
- Es gibt Kanten (A, B) für alle $A, B \in N_2$, so dass $A \rightarrow B$ eine Regel von \mathcal{G}_2 ist, und Kanten (A, α) für alle $A \in N_2$ und Satzformen $\alpha \notin N_2$, so dass $A \rightarrow \alpha$ eine Regel von \mathcal{G}_2 ist.

Dann gibt es für $A_1 \in N_2$ und $\alpha \in (N \cup \Sigma^*) \setminus N_2$ genau dann eine Ableitung $A_1 \rightarrow_{\mathcal{G}_2} A_2 \rightarrow_{\mathcal{G}_2} \dots \rightarrow_{\mathcal{G}_2} A_n \rightarrow_{\mathcal{G}_2} \alpha$ mit $n \geq 2$ und $A_2, \dots, A_n \in N_2$, wenn es einen Weg in D von A_1 nach α gibt.

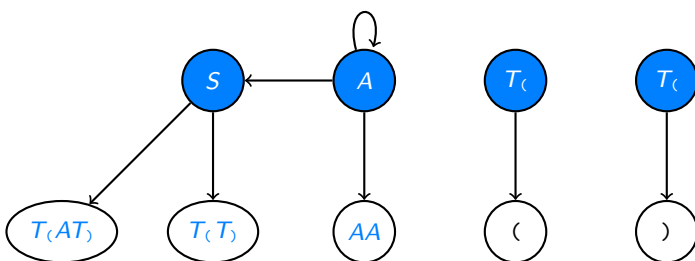
Um \mathcal{G}_3 zu konstruieren müssen wir also eine Regel $A \rightarrow \alpha$ für alle $A \in N_2$ und alle $\alpha \in (N \cup \Sigma^*) \setminus N_2$ hinzufügen, für die ein Weg von A nach α in D existiert.

Beispiel 6.22

Aus der Grammatik \mathcal{G}_K für die Sprache L_K der korrekten Klammerausdrücke erhalten wir in den Schritten 1 und 2 folgende Grammatik \mathcal{G}_2 :

$$\begin{aligned} S &\rightarrow T_1 A T_1 \mid T_1 T_1 \\ A &\rightarrow S \mid A A \mid A \\ T_1 &\rightarrow (\\ T_1 &\rightarrow) \end{aligned}$$

Für diese Grammatik sieht der Graph D wie folgt aus:



Wir löschen die Regeln $A \rightarrow S$ und $A \rightarrow A$ und fügen neue Regeln $A \rightarrow T_1 A T_1$ und $A \rightarrow T_1 T_1$ hinzu. Also erhalten wir die Grammatik

$$\begin{aligned} S &\rightarrow T_1 A T_1 \mid T_1 T_1 \\ A &\rightarrow T_1 A T_1 \mid T_1 T_1 \mid A A \\ T_1 &\rightarrow (\\ T_1 &\rightarrow) \end{aligned}$$

Als Ergebnis von Schritt 3 haben wir eine kontextfreie Grammatik

$$\mathcal{G}_3 = (N_3, \Sigma, P_3, S_3)$$

die nur Regeln der Formen

$$A \rightarrow a \quad \text{und} \quad A \rightarrow A_1 \dots A_m$$

mit $A, A_1, \dots, A_m \in N$, $m \geq 2$ und $a \in \Sigma$ enthält.

Ziel

Konstruktion einer zu \mathcal{G}_3 äquivalenten Grammatik \mathcal{G}_4 , die nur Regeln der Formen

$$A \rightarrow a \quad \text{und} \quad A \rightarrow A_1 A_2$$

mit $A, A_1, A_2 \in N$ und $a \in \Sigma$ enthält

Idee

$$\curvearrowright A \rightarrow A_1 B_1, B_1 \rightarrow A_2 \dots A_m$$

Ersetze Regel $A \rightarrow A_1 \dots A_m$ mit $m \geq 2$ durch

$$A \rightarrow A_1 B_1, \quad B_1 \rightarrow A_2 B_2, \quad B_2 \rightarrow A_3 B_3, \\ \dots \quad B_{m-3} \rightarrow A_{m-2} B_{m-2}, \quad B_{m-2} \rightarrow A_{m-1} A_m$$

mit neuen Nichtterminalsymbolen B_i .

Beispiel 6.23

Aus der Grammatik \mathcal{G}_K für die Sprache L_K der korrekten Klammerausdrücke erhalten wir in den Schritten 1–3 folgende Grammatik \mathcal{G}_3 :

$$\begin{aligned} S &\rightarrow T_\epsilon(AT_\epsilon) \mid T_\epsilon T_\epsilon \\ A &\rightarrow T_\epsilon(AT_\epsilon) \mid T_\epsilon T_\epsilon \mid AA \\ T_\epsilon &\rightarrow (\\ T_\epsilon &\rightarrow) \end{aligned}$$

Wir müssen die beiden Regeln $S \rightarrow T_\epsilon(AT_\epsilon)$ und $A \rightarrow T_\epsilon(AT_\epsilon)$ ersetzen durch

$$S \rightarrow T_\epsilon B_1, \quad B_1 \rightarrow AT_\epsilon$$

bzw.

$$A \rightarrow T_\epsilon C_1, \quad C_1 \rightarrow AT_\epsilon.$$

Insgesamt erhalten wir folgende Grammatik \mathcal{G}_4 in CNF:

$$\begin{aligned} S &\rightarrow T_\epsilon B_1 \mid T_\epsilon T_\epsilon & B_1 &\rightarrow AT_\epsilon \\ A &\rightarrow T_\epsilon C_1 \mid T_\epsilon T_\epsilon \mid AA & C_1 &\rightarrow AT_\epsilon \\ T_\epsilon &\rightarrow (\\ T_\epsilon &\rightarrow) \end{aligned}$$

Konstruktion

Wir bilden die Grammatik \mathcal{G}_4 aus \mathcal{G}_3 wie folgt:

- Für jede Regel ρ der Gestalt $A \rightarrow A_1 \dots A_m$ mit $m \geq 2$ führen wir neue Nichtterminalsymbole $B_1^\rho, \dots, B_{m-2}^\rho$ ein.
- Wir fügen die $m - 1$ neuen Regeln

$$A \rightarrow A_1 B_1^\rho, \quad B_1^\rho \rightarrow A_2 B_2^\rho, \quad B_2^\rho \rightarrow A_3 B_3^\rho, \\ \dots \quad B_{m-3}^\rho \rightarrow A_{m-2} B_{m-2}^\rho, \quad B_{m-2}^\rho \rightarrow A_{m-1} A_m$$

hinzu.

- Wir löschen die Regel ρ .

Dann lässt sich leicht beweisen, dass $L(\mathcal{G}_4) = L(\mathcal{G}_3)$.

Definition 6.24

Eine kontextfreie Grammatik ist in **Greibach-Normalform**, wenn sie nur Regeln der Form

$$A \rightarrow aB_1 \dots B_k,$$

für $k \geq 0$, Nichtterminalsymbole A, B_1, \dots, B_k und Terminalsymbole a hat.

Definition 6.24

Eine kontextfreie Grammatik ist in **Greibach-Normalform**, wenn sie nur Regeln der Form

$$A \rightarrow aB_1 \dots B_k,$$

für $k \geq 0$, Nichtterminalsymbole A, B_1, \dots, B_k und Terminalsymbole a hat.

Satz 6.25

Jede kontextfreie Grammatik ist äquivalent zu einer kontextfreien Grammatik in Greibach-Normalform.

Definition 6.24

Eine kontextfreie Grammatik ist in **Greibach-Normalform**, wenn sie nur Regeln der Form

$$A \rightarrow aB_1 \dots B_k,$$

für $k \geq 0$, Nichtterminalsymbole A, B_1, \dots, B_k und Terminalsymbole a hat.

Satz 6.25

Jede kontextfreie Grammatik ist äquivalent zu einer kontextfreien Grammatik in Greibach-Normalform.

Genauer: Zu jeder kontextfreien Grammatik G gibt es eine kontextfreie Grammatik G' in Greibach-Normalform, so dass $L(G') = L(G) \setminus \{\varepsilon\}$.

Definition 6.24

Eine kontextfreie Grammatik ist in **Greibach-Normalform**, wenn sie nur Regeln der Form

$$A \rightarrow aB_1 \dots B_k,$$

für $k \geq 0$, Nichtterminalsymbole A, B_1, \dots, B_k und Terminalsymbole a hat.

Satz 6.25

Jede kontextfreie Grammatik ist äquivalent zu einer kontextfreien Grammatik in Greibach-Normalform.

Genauer: Zu jeder kontextfreien Grammatik G gibt es eine kontextfreie Grammatik G' in Greibach-Normalform, so dass $L(G') = L(G) \setminus \{\varepsilon\}$.

(Ohne Beweis.)

Beispiel 6.26

Sei \mathcal{G}'_K die kontextfreie Grammatik

$$\begin{aligned} S &\rightarrow (A) \mid () \\ A &\rightarrow S \mid AA \end{aligned}$$

die die Sprache $L_K \setminus \{\varepsilon\}$ erzeugt.

Beispiel 6.26

Sei \mathcal{G}'_K die kontextfreie Grammatik

$$\begin{aligned} S &\rightarrow (A) \mid () \\ A &\rightarrow S \mid AA \end{aligned}$$

die die Sprache $L_K \setminus \{\varepsilon\}$ erzeugt.

Eine äquivalente Grammatik in Greibach-Normalform ist

$$\begin{aligned} S &\rightarrow (AT_1 \mid (T_1 \\ A &\rightarrow (AT_1 \mid (T_1 \mid (AT_1A \mid (T_1A \\ T_1 &\rightarrow) \end{aligned}$$

Beispiel 6.26

Sei \mathcal{G}'_K die kontextfreie Grammatik

$$\begin{aligned} S &\rightarrow (A) \mid () \\ A &\rightarrow S \mid AA \end{aligned}$$

die die Sprache $L_K \setminus \{\varepsilon\}$ erzeugt.

Eine äquivalente Grammatik in Greibach-Normalform ist

$$\begin{aligned} S &\rightarrow (AT_j \mid (T_j \\ A &\rightarrow (AT_j \mid (T_j \mid (AT_jA \mid (T_jA \\ T_j &\rightarrow) \end{aligned}$$

Beweisidee für Satz 6.25

Wir verfolgen „linkestmögliche“ Ableitungen, bis ganz links ein Terminalsymbol steht, und bilden daraus neue Regeln. Für weiter rechts auftretende Terminalsymbole verwenden wir den gleichen Trick wie bei der Chomsky-Normalform.

Definition 6.27

Eine kontextfreie Grammatik ist **rechtslinear**, wenn sie nur Regeln der Form

$$A \rightarrow aB$$

$$A \rightarrow a,$$

für Nichtterminalsymbole A, B , und Terminalsymbole a , hat.

Definition 6.27

Eine kontextfreie Grammatik ist **rechtslinear**, wenn sie nur Regeln der Form

$$A \rightarrow aB$$

$$A \rightarrow a,$$

für Nichtterminalsymbole A, B , und Terminalsymbole a , hat.

Beobachtung 6.28

Jede rechtslineare Grammatik ist in Greibach-Normalform, aber nicht umgekehrt.

Definition 6.27

Eine kontextfreie Grammatik ist **rechtslinear**, wenn sie nur Regeln der Form

$$A \rightarrow aB$$

$$A \rightarrow a,$$

für Nichtterminalsymbole A, B , und Terminalsymbole a , hat.

Beobachtung 6.28

Jede rechtslineare Grammatik ist in Greibach-Normalform, aber nicht umgekehrt.

Achtung

Rechtslineare Grammatiken bilden keine Normalform!

Es gibt kontextfreie Sprachen (die ε nicht enthalten), für die es keine rechtslineare Grammatik gibt.

(Das folgt aus dem nächsten Satz.)

Satz 6.29

Rechtslineare Grammatiken erzeugen gerade die regulären Sprachen.

Satz 6.29

Rechtslineare Grammatiken erzeugen gerade die regulären Sprachen.

Genauer:

1. *Zu jeder regulären Sprache L gibt es eine rechtslineare kontextfreie Grammatik \mathcal{G} so dass $L(\mathcal{G}) = L \setminus \{\varepsilon\}$.*

Satz 6.29

Rechtslineare Grammatiken erzeugen gerade die regulären Sprachen.

Genauer:

1. *Zu jeder regulären Sprache L gibt es eine rechtslineare kontextfreie Grammatik \mathcal{G} so dass $L(\mathcal{G}) = L \setminus \{\varepsilon\}$.*
2. *Für jede rechtslineare kontextfreie Grammatik \mathcal{G} ist die Sprache $L(\mathcal{G})$ regulär.*

Behauptung 1

Zu jeder regulären Sprache L gibt es eine rechtslineare kontextfreie Grammatik \mathcal{G} so dass $L(\mathcal{G}) = L \setminus \{\varepsilon\}$.

Beweis.

Unser Beweis, dass jede reguläre Sprache kontextfrei ist, liefert zu jedem NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ folgende äquivalente kontextfreie Grammatik $\mathcal{G}_{\mathcal{A}} = (Q, \Sigma, P, q_0)$ mit Regeln

$$\begin{aligned} q &\rightarrow ar && \text{für alle } (q, a, r) \in \Delta, \\ q &\rightarrow \varepsilon && \text{für alle } q \in F. \end{aligned}$$

Wir eliminieren die ε -Regeln, indem wir für alle $(q, a, r) \in \Delta$ mit $r \in F$ die Regel $q \rightarrow a$ hinzufügen (und danach die Regeln $q \rightarrow \varepsilon$ entfernen).

Die resultierende Grammatik ist rechtslinear. □

Behauptung 2

Für jede rechtslineare kontextfreie Grammatik \mathcal{G} ist die Sprache $L(\mathcal{G})$ regulär.

Beweis.

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine rechtslineare kontextfreie Grammatik.

Wir beobachten zunächst, dass jede Ableitung eines Terminalwortes in \mathcal{G} die Form

$$S \rightarrow a_1 A_1 \rightarrow a_1 a_2 A_2 \rightarrow \dots \rightarrow a_1 \dots a_{k-1} A_{k-1} \rightarrow a_1 \dots a_k \quad (\star)$$

für ein $k \geq 0$ und Regeln $S \rightarrow a_1 A_1$, $A_1 \rightarrow a_2 A_2$, $A_2 \rightarrow a_3 A_3$, ..., $A_{k-2} \rightarrow a_{k-1} A_{k-1}$, $A_{k-1} \rightarrow a_k$ hat.

Wir definieren einen NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ durch

- ▶ $Q := N \cup \{q_f\}$, wobei $q_f \notin N$,
- ▶ $\Delta = \{(A, a, B) \mid A \rightarrow aB \in P\} \cup \{(A, a, q_f) \mid A \rightarrow a \in P\}$,
- ▶ $q_0 := S$,
- ▶ $F := \{q_f\}$.

Dann entspricht die Ableitung (\star) des Wortes $a_1 \dots a_k \in L(\mathcal{G})$ dem akzeptierenden Lauf

$$S \xrightarrow{a_1} A_1 \xrightarrow{a_2} A_2 \xrightarrow{a_3} \dots \xrightarrow{a_{k-1}} A_{k-1} \xrightarrow{a_k} q_f. \quad (\star\star)$$

Also gilt $L(\mathcal{G}) \subseteq L(\mathcal{A})$.

Umgekehrt hat jeder akzeptierende Lauf von \mathcal{A} auf einem Wort $w = a_1 \dots a_k$ die Gestalt $(\star\star)$ und entspricht damit einer Ableitung von w in \mathcal{G} der Gestalt (\star) . Also gilt auch $L(\mathcal{A}) \subseteq L(\mathcal{G})$. □

Wir betrachten folgende rechtslineare Grammatik:

$$S \rightarrow 0S \mid 1A \mid 0$$

$$A \rightarrow 0B$$

$$B \rightarrow 0S \mid 1A \mid 0$$

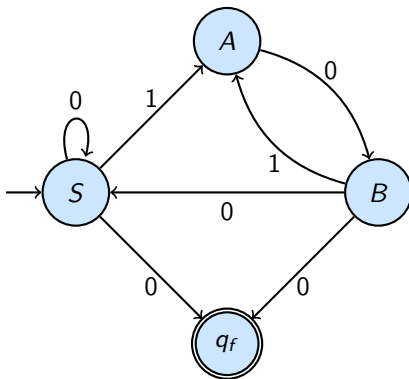
Wir betrachten folgende rechtslineare Grammatik:

$$S \rightarrow 0S \mid 1A \mid 0$$

$$A \rightarrow 0B$$

$$B \rightarrow 0S \mid 1A \mid 0$$

Unsere Konstruktion liefert folgenden äquivalenten NFA:



Abschnitt 6.3

Ableitungsbäume

Ableitungen sind eigentlich nicht linear, sondern haben eine Baumstruktur, denn die Reihenfolge, in der die Regeln angewandt werden, ist unwichtig.

Ableitungen sind eigentlich nicht linear, sondern haben eine Baumstruktur, denn die Reihenfolge, in der die Regeln angewandt werden, ist unwichtig.

Beispiel 6.31

Die Ableitungen

$$S \rightarrow AB \rightarrow \alpha B \rightarrow \alpha\beta$$

und

$$S \rightarrow AB \rightarrow A\beta \rightarrow \alpha\beta$$

sind „im Wesentlichen gleich“.

Ableitungen sind eigentlich nicht linear, sondern haben eine Baumstruktur, denn die Reihenfolge, in der die Regeln angewandt werden, ist unwichtig.

Beispiel 6.31

Die Ableitungen

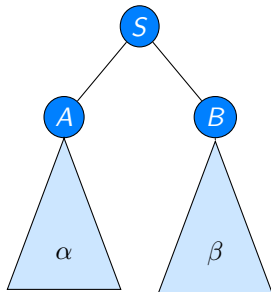
$$S \rightarrow AB \rightarrow \alpha B \rightarrow \alpha\beta$$

und

$$S \rightarrow AB \rightarrow A\beta \rightarrow \alpha\beta$$

sind „im Wesentlichen gleich“.

Für viele Zwecke geeigneter ist die Darstellung der beiden Ableitungen als Baum.



Grammatik:

$$\begin{aligned} S &\rightarrow aAb \mid ab \\ A &\rightarrow S \mid AA \end{aligned}$$

Grammatik:

$$\begin{aligned} S &\rightarrow aAb \mid ab \\ A &\rightarrow S \mid AA \end{aligned}$$

Eine Ableitung:

$$\begin{aligned} \underline{S} &\rightarrow a\underline{A}b \rightarrow a\underline{AA}b \rightarrow a\underline{AAA}b \rightarrow a\underline{S}AAb \rightarrow aab\underline{AA}b \rightarrow aab\underline{S}Ab \\ &\rightarrow aabab\underline{A}b \rightarrow aabab\underline{S}b \rightarrow aababa\underline{A}bb \rightarrow aababa\underline{S}bb \rightarrow aababaabbb \end{aligned}$$

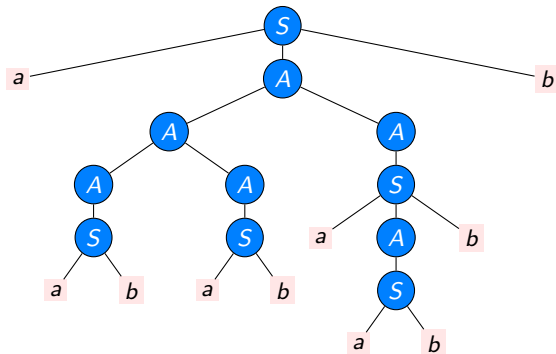
Grammatik:

$$\begin{aligned} S &\rightarrow aAb \mid ab \\ A &\rightarrow S \mid AA \end{aligned}$$

Eine Ableitung:

$$\begin{aligned} \underline{S} &\rightarrow a\underline{A}b \rightarrow a\underline{AA}b \rightarrow a\underline{AAA}b \rightarrow a\underline{SAA}b \rightarrow aab\underline{AAb} \rightarrow aab\underline{SAb} \\ &\rightarrow aabab\underline{Ab} \rightarrow aabab\underline{S}b \rightarrow aababa\underline{Abb} \rightarrow aababa\underline{S}bb \rightarrow aababaabbbb \end{aligned}$$

Ableitungsbaum:



Format der Ableitungsbäume

Wir unterscheiden in einem Ableitungsbaum T zu einer Grammatik $\mathcal{G} = (N, \Sigma, P, S)$

- ▶ die **Knotenmenge** $\text{dom}(T)$
- ▶ die **Beschriftungsfunktion**

$$T : \text{dom}(T) \rightarrow (N \cup \Sigma),$$

die jedem Knoten ein Symbol aus N oder Σ zuordnet

Definition 6.33

1. Ein **Baumbereich** ist eine Menge $D \subseteq (\mathbb{N}_+)^*$, die
 - ▶ unter Präfixbildung abgeschlossen ist,
 - ▶ jeweils für uj auch alle ui mit $1 \leq i \leq j$ enthält.

In dieser Vorlesung betrachten wir nur endliche Baumbereiche.

Definition 6.33

1. Ein **Baumbereich** ist eine Menge $D \subseteq (\mathbb{N}_+)^*$, die
 - ▶ unter Präfixbildung abgeschlossen ist,
 - ▶ jeweils für uj auch alle ui mit $1 \leq i \leq j$ enthält.

In dieser Vorlesung betrachten wir nur endliche Baumbereiche.

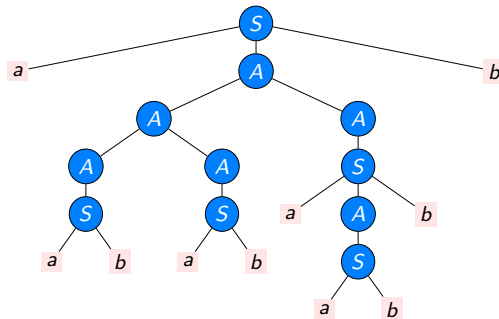
2. Sei Σ ein Alphabet. Ein **Σ -Baum** ist eine Funktion $T : \text{dom}(T) \rightarrow \Sigma$, wobei $\text{dom}(T)$ ein Baumbereich ist.

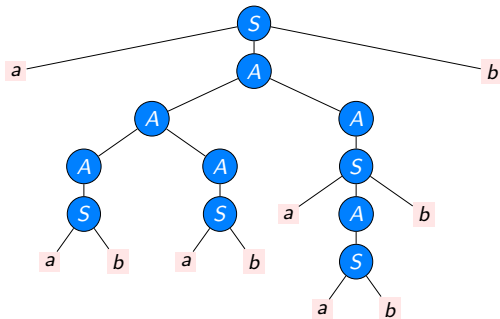
Definition 6.33

1. Ein **Baubereich** ist eine Menge $D \subseteq (\mathbb{N}_+)^*$, die
 - ▶ unter Präfixbildung abgeschlossen ist,
 - ▶ jeweils für uj auch alle ui mit $1 \leq i \leq j$ enthält.

In dieser Vorlesung betrachten wir nur endliche Baubereiche.

2. Sei Σ ein Alphabet. Ein **Σ -Baum** ist eine Funktion $T : \text{dom}(T) \rightarrow \Sigma$, wobei $\text{dom}(T)$ ein Baubereich ist.
3. Ein Baum T ist **höchstens k -verzweigt**, wenn $\text{dom}(T) \subseteq \{1, \dots, k\}^*$.

Ableitungsbaum T 

Ableitungsbaum T 

Knotenmenge

$$\text{dom}(T) = \{\varepsilon, 1, 2, 21, 211, 2111, 21111, 21112, 212, 2121, 21211, 21212, 22, 221, 2211, 2212, 22121, 221211, 221212, 2213, 3\}$$

Beschriftung

v	ε	1	2	21	211	2111	21111	21112	212	2121	\dots
$T(v)$	S	a	A	A	A	S	a	b	A	S	\dots

Sei T ein Σ -Baum.

- ▶ ε ist die **Wurzel** von T .

Sei T ein Σ -Baum.

- ▶ ε ist die **Wurzel** von T .
- ▶ Ein Knoten $ui \in \text{dom}(T)$ ist *iter* **Nachfolger** (oder *ites* **Kind**) von u ,
umgekehrt ist u der **Vorgänger** von ui .

Sei T ein Σ -Baum.

- ▶ ε ist die **Wurzel** von T .
- ▶ Ein Knoten $ui \in \text{dom}(T)$ ist *iter* **Nachfolger** (oder *ites* **Kind**) von u , umgekehrt ist u der **Vorgänger** von ui .
- ▶ Die **Blätter** von T sind die Knoten ohne Nachfolger, alle andere Knoten sind **innere Knoten**.

Sei T ein Σ -Baum.

- ▶ ε ist die **Wurzel** von T .
- ▶ Ein Knoten $ui \in \text{dom}(T)$ ist *iter* **Nachfolger** (oder *ites* **Kind**) von u , umgekehrt ist u der **Vorgänger** von ui .
- ▶ Die **Blätter** von T sind die Knoten ohne Nachfolger, alle andere Knoten sind **innere Knoten**.
- ▶ Seien $b_1, \dots, b_n \in \text{dom}(T)$ die Blätter von T in lexikographischer Reihenfolge, d.h., von links nach rechts. Dann ist das Wort

$$\beta(T) := T(b_1) \dots T(b_n) \in \Sigma^*$$

die **Blattbeschriftung** von T .

Sei T ein Σ -Baum.

- ▶ Ein **Ast** (der Länge n) in T ist eine Knotenfolge v_0, \dots, v_n mit $v_0 = \varepsilon$, v_{i+1} jeweils Nachfolger von v_i , und Blatt v_n ,

Sei T ein Σ -Baum.

- ▶ Ein **Ast** (der Länge n) in T ist eine Knotenfolge v_0, \dots, v_n mit $v_0 = \varepsilon$, v_{i+1} jeweils Nachfolger von v_i , und Blatt v_n ,
- ▶ die **Höhe** $h(T)$ des Baumes T ist die Länge eines längsten Astes, also

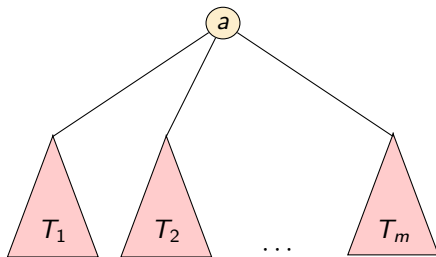
$$h(T) := \max\{|v| \mid v \in \text{dom}(T)\}.$$

Zerlegung an der Wurzel

Ein Baum T der Höhe $h + 1$ mit Wurzelbeschriftung a hat die Form

Zerlegung an der Wurzel

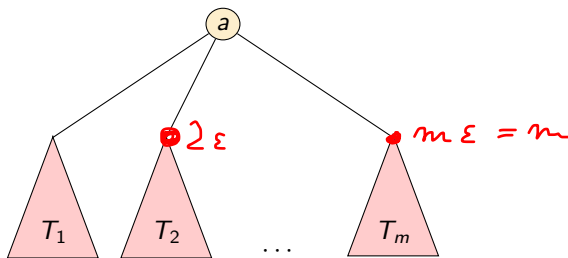
Ein Baum T der Höhe $h + 1$ mit Wurzelbeschriftung a hat die Form



mit Bäumen T_1, \dots, T_m der Höhe jeweils $\leq h$,

Zerlegung an der Wurzel

Ein Baum T der Höhe $h + 1$ mit Wurzelbeschriftung a hat die Form



mit Bäumen T_1, \dots, T_m der Höhe jeweils $\leq h$,
d.h. es gilt

- ▶ $\text{dom}(T) = \{\varepsilon\} \cup \{iv \mid 1 \leq i \leq m, v \in v(T_i)\},$
- ▶ $T(\varepsilon) = a, T(1v) = T_1(v), \dots, T(mv) = T_m(v).$

Definition 6.34

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine Grammatik.

1. Ein **Ableitungsbaum zu \mathcal{G}** (englisch: **Parse Tree**) ist ein $(N \cup \Sigma)$ -Baum T , für den gilt:
 - ▶ $T(\varepsilon) \in N$,
 - ▶ hat v die Nachfolger $v1, \dots, vm$ in $\text{dom}(T)$, so ist $T(v) \rightarrow T(v1) \dots T(vm)$ eine Regel in P .

Definition 6.34

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine Grammatik.

1. Ein **Ableitungsbaum zu \mathcal{G}** (englisch: **Parse Tree**) ist ein $(N \cup \Sigma)$ -Baum T , für den gilt:
 - ▶ $T(\varepsilon) \in N$,
 - ▶ hat v die Nachfolger v_1, \dots, v_m in $\text{dom}(T)$, so ist $T(v) \rightarrow T(v_1) \dots T(v_m)$ eine Regel in P .
2. Ein Ableitungsbaum T zu \mathcal{G} ist **vollständig**, wenn $T(\varepsilon) = S$ und $\beta(T) \in \Sigma^*$.

Satz 6.35

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Dann gilt für alle $A \in N$ und $\alpha \in (N \cup \Sigma)^*$:

$A \xrightarrow{*} \alpha \iff$ es gibt einen Ableitungsbaum T zu \mathcal{G} mit $T(\varepsilon) = A$ und $\beta(T) = \alpha$.

Satz 6.35

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Dann gilt für alle $A \in N$ und $\alpha \in (N \cup \Sigma)^*$:

$$A \xrightarrow{*} \alpha \iff \text{es gibt einen Ableitungsbaum } T \text{ zu } \mathcal{G} \text{ mit} \\ T(\varepsilon) = A \text{ und } \beta(T) = \alpha.$$

Wir nennen T auch einen *Ableitungsbaum für α aus A in \mathcal{G}*

Satz 6.35

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Dann gilt für alle $A \in N$ und $\alpha \in (N \cup \Sigma)^*$:

$$A \xrightarrow{*} \alpha \iff \text{es gibt einen Ableitungsbaum } T \text{ zu } \mathcal{G} \text{ mit} \\ T(\varepsilon) = A \text{ und } \beta(T) = \alpha.$$

Wir nennen T auch einen **Ableitungsbaum für α aus A in \mathcal{G}**

Korollar 6.36

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Dann gilt für alle $w \in \Sigma^*$:

$$w \in L(\mathcal{G}) \iff \text{es gibt einen vollständigen Ableitungsbaum } T \text{ zu} \\ \mathcal{G} \text{ mit } \beta(T) = w.$$

Ableitungen und Ableitungsbäume

Satz 6.35

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Dann gilt für alle $A \in N$ und $\alpha \in (N \cup \Sigma)^*$:

$$A \xrightarrow{*} \alpha \iff \text{es gibt einen Ableitungsbaum } T \text{ zu } \mathcal{G} \text{ mit} \\ T(\varepsilon) = A \text{ und } \beta(T) = \alpha.$$

Wir nennen T auch einen **Ableitungsbaum für α aus A in \mathcal{G}**

Korollar 6.36

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik. Dann gilt für alle $w \in \Sigma^*$:

$$w \in L(\mathcal{G}) \iff \text{es gibt einen vollständigen Ableitungsbaum } T \text{ zu} \\ \mathcal{G} \text{ mit } \beta(T) = w.$$

Wir nennen T auch einen **Ableitungsbaum für w in \mathcal{G}**

Wir betrachten wieder die Grammatik

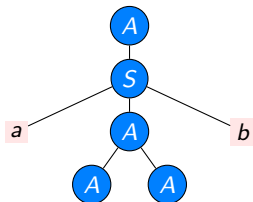
$$\begin{aligned} S &\rightarrow aAb \mid ab \\ A &\rightarrow S \mid AA \end{aligned}$$

Wir betrachten wieder die Grammatik

$$S \rightarrow aAb \mid ab$$

$$A \rightarrow S \mid AA$$

Ableitungsbaum für $aAAb$ aus A :

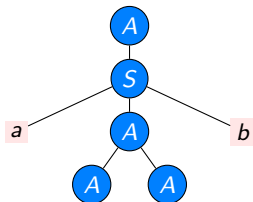


Beispiele 6.37

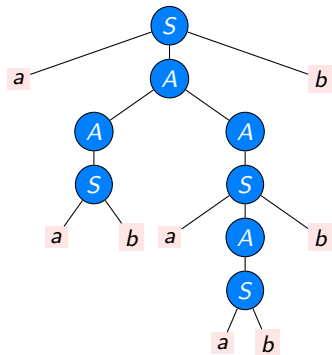
Wir betrachten wieder die Grammatik

$$\begin{aligned} S &\rightarrow aAb \mid ab \\ A &\rightarrow S \mid AA \end{aligned}$$

Ableitungsbaum für $aAAb$ aus A :



Ableitungsbaum für $aabaabbb$:



Satz 6.35

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik und $A \in N$ und $\alpha \in (N \cup \Sigma)^*$. Dann gilt

$$A \xrightarrow{*} \alpha \iff \text{es gibt einen Ableitungsbaum } T \text{ zu } \mathcal{G} \text{ mit } T(\varepsilon) = A \text{ und } \beta(T) = \alpha.$$

Beweis.

„ \implies “: Induktion über die Länge n einer Ableitung von α aus A in \mathcal{G} .

Induktionsschritt $n = 0$:

Gelte $A \xrightarrow{0} \alpha$. Dann gilt $\alpha = A$, und



ist ein Ableitungsbaum für α aus A .

Induktionsschritt $0, \dots, n \rightarrow n + 1$:

Sei

$$\alpha_0 = A \rightarrow \alpha_1 \rightarrow \dots \rightarrow \alpha_{n+1} = \alpha$$

eine Ableitung von α aus A .

Wir zerlegen α_1 in Terminalwörter und Nichtterminale:

$$\alpha_1 = w_0 B_1 w_1 B_2 \dots w_{k-1} B_k w_k,$$

für ein $k \geq 0$, $w_0, \dots, w_k \in \Sigma^*$ und $B_1, \dots, B_k \in N$. Die w_i s können das leere Wort sein.

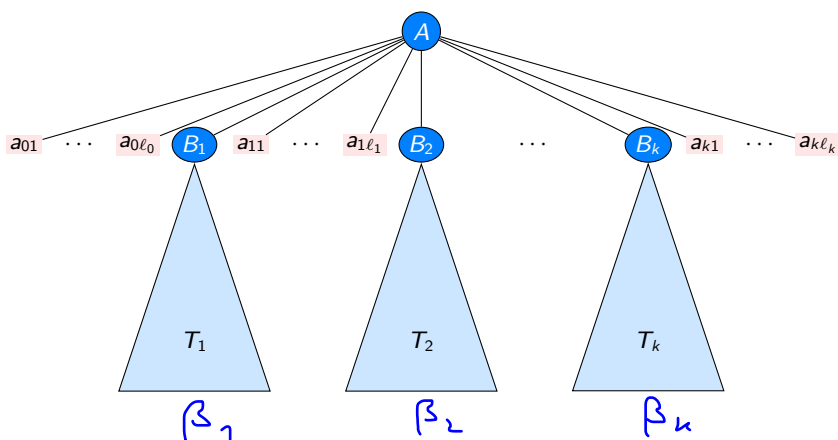
Nach dem Zerlegungslemma 6.10 lässt sich α dann zerlegen als

$$\alpha = w_0 \beta_1 w_1 \beta_2 \dots w_{k-1} \beta_k w_k,$$

so dass $B_i \xrightarrow{\leq n} \beta_i$ für $1 \leq i \leq k$.

Nach Induktionsannahme gibt es für $1 \leq i \leq k$ einen Ableitungsbaum T_i für β_i aus B_i .

Nehmen wir an, für $0 \leq i \leq k$ ist $w_i = a_{i1} \dots a_{i\ell_i}$. Dann ist



ein Ableitungsbaum für α aus A .

„ \impliedby “: Induktion über die Höhe h eines Ableitungsbaums T für α aus A .

Induktionsanfang $h = 0$:

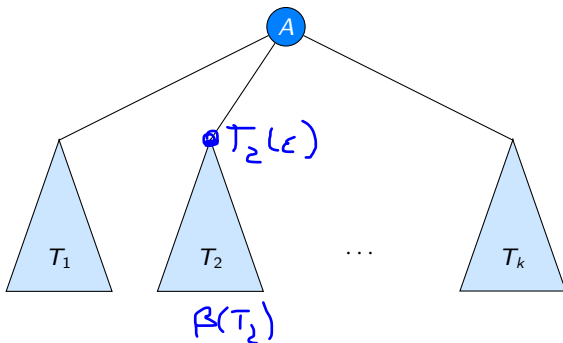
Ein Ableitungsbaum der Höhe 0 von α aus A hat die Gestalt



Dann gilt $\alpha = A$, und es gilt $A \xrightarrow{0} \alpha$.

Induktionsschritt $0, \dots, h \rightarrow h + 1$:

Sei T ein Ableitungsbaum der Höhe $h + 1$ von α aus A . Wir zerlegen T an der Wurzel in Bäume T_1, \dots, T_k der Höhe $\leq h$:



Dann gilt

$$\alpha = \beta(T) = \beta(T_1) \dots \beta(T_k).$$

Außerdem ist

$$A \rightarrow T_1(\varepsilon) \dots T_k(\varepsilon)$$

eine Regel von \mathcal{G} .

Nach Induktionsannahme gilt für $1 \leq i \leq k$

$$T_i(\varepsilon) \xrightarrow{*} \beta(T_i).$$

Nach dem Kombinationslemma 6.9 gilt dann

$$A \rightarrow T_1(\varepsilon) \dots T_k(\varepsilon) \xrightarrow{*} \beta(T_1) \dots \beta(T_k) = \alpha.$$

□

Wörter mit einkodierten Ableitungsbäumen

Idee

Wir wollen Ableitungsbäume „linearisieren“ und sie wieder als Wörter kodieren.

Wörter mit einkodierten Ableitungsbäumen

Idee

Wir wollen Ableitungsbäume „linearisieren“ und sie wieder als Wörter kodieren.

Genau das geschieht in XML-Dokumenten.

Wörter mit einkodierten Ableitungsbäumen

Idee

Wir wollen Ableitungsbäume „linearisieren“ und sie wieder als Wörter kodieren.

Genau das geschieht in XML-Dokumenten.

Methode

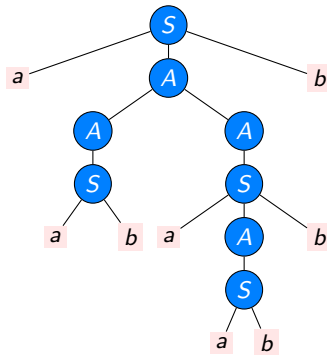
Für jedes Nichtterminalsymbol A führen wir neue Terminalsymbole

$\langle A$ und $\rangle A$

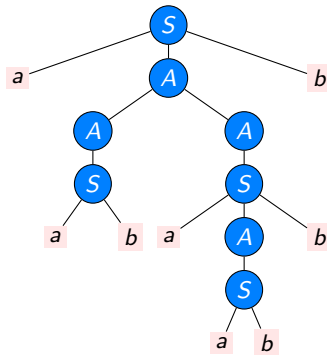
ein.

Ableitungsbaum T für ein Wort w wird durch ein Wort $x(T)$ kodiert, das aus w durch Einfügen dieser neuen Symbole entsteht.

Ableitungsbaum T



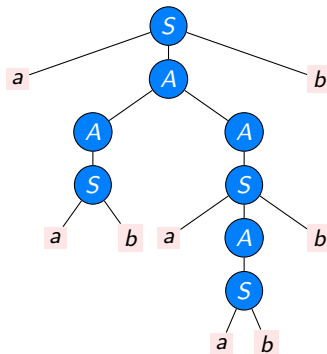
Ableitungsbaum T



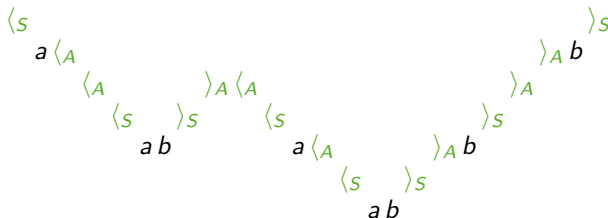
Wort $x(T)$ mit einkodiertem Ableitungsbaum

$\langle s a \langle A \langle A \langle S a b \rangle S \rangle A \langle A \langle S a \langle A \langle S a b \rangle S \rangle A b \rangle S \rangle A \rangle A b \rangle S$

Ableitungsbaum T



Wort $x(T)$ mit einkodiertem Ableitungsbaum



Definition der Baumkodierung

Definition 6.38

Sei $\mathcal{G} = (N, \Sigma, P, S)$.

Für jeden Ableitungsbaum T zu \mathcal{G} definieren wir rekursiv über die Höhe h von T ein Wort

$$x(T) \in \left(\Sigma \cup \{ \langle A, \rangle_A \mid A \in N \} \right)^*.$$

Definition der Baumkodierung

Definition 6.38

Sei $\mathcal{G} = (N, \Sigma, P, S)$.

Für jeden Ableitungsbaum T zu \mathcal{G} definieren wir rekursiv über die Höhe h von T ein Wort

$$x(T) \in \left(\Sigma \cup \{ \langle A, \rangle_A \mid A \in N \} \right)^*.$$

Rekursionsanfang $h = 0$:

$$x(T) := \begin{cases} a & \text{falls } T(\varepsilon) = a \in \Sigma, \\ \langle A \rangle_A & \text{falls } T(\varepsilon) = A \in N. \end{cases}$$

Definition der Baumkodierung

Definition 6.38

Sei $\mathcal{G} = (N, \Sigma, P, S)$.

Für jeden Ableitungsbaum T zu \mathcal{G} definieren wir rekursiv über die Höhe h von T ein Wort

$$x(T) \in \left(\Sigma \cup \{ \langle A \rangle_A \mid A \in N \} \right)^*.$$

Rekursionsanfang $h = 0$:

$$x(T) := \begin{cases} a & \text{falls } T(\varepsilon) = a \in \Sigma, \\ \langle A \rangle_A & \text{falls } T(\varepsilon) = A \in N. \end{cases}$$

Rekursionsschritt $0, \dots, h \rightarrow h + 1$:

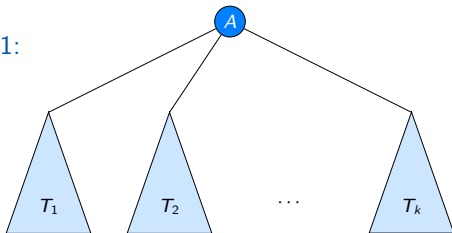
Wir zerlegen T an der Wurzel.

Sei $T(\varepsilon) = A$, und seien

T_1, \dots, T_k die Teilbäume.

Wir setzen

$$x(T) := \langle A x(T_1) \dots x(T_k) \rangle_A.$$



XML-Dokumente sind Wörter der Form $x(T)$ mit modifizierter Klammernotation:

- ▶ $\langle A \rangle$ für $\langle A$,
- ▶ $\langle /A \rangle$ für \rangle_A .

XML-Dokumente sind Wörter der Form $x(T)$ mit modifizierter Klammernotation:

- ▶ $\langle A \rangle$ für $\langle A$,
- ▶ $\langle /A \rangle$ für $\rangle A$.

Beispiel 6.39

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Studienprofil [...]>
<Studienprofil>
  <Studiengang> INFORMATIK </Studiengang>
  <Fachsemester> 3 </Fachsemester>
  <Vorlesungen>
    <Titel> PROGRAMMIERUNG </Titel>
    <Titel> TECHNISCHE INFORMATIK </Titel>
    <Titel> DATENSTRUKTUREN </Titel>
  </Vorlesungen>
</Studienprofil>
```

Document-Type Definitionen (DTD's)

Eine **Document-Type Definition (DTD)** ist ein Regelsystem zur Erzeugung einer Menge von XML-Dokumenten, also eine Grammatik zum Aufbau von Terminalwörtern **zusammen mit zugehörigen Ableitungsbäumen**.

Document-Type Definitionen (DTD's)

Eine **Document-Type Definition (DTD)** ist ein Regelsystem zur Erzeugung einer Menge von XML-Dokumenten, also eine Grammatik zum Aufbau von Terminalwörtern **zusammen mit zugehörigen Ableitungsbäumen**.

Beispiel 6.40

DTD dargestellt als verallgemeinerte kontextfreie Grammatik.

[Studienprofil](#) → [Studiengang](#) · [Fachsemester](#) · [Vorlesungen](#)

Document-Type Definitionen (DTD's)

Eine **Document-Type Definition (DTD)** ist ein Regelsystem zur Erzeugung einer Menge von XML-Dokumenten, also eine Grammatik zum Aufbau von Terminalwörtern **zusammen mit zugehörigen Ableitungsbäumen**.

Beispiel 6.40

DTD dargestellt als verallgemeinerte kontextfreie Grammatik.

Studienprofil → **Studiengang** · **Fachsemester** · **Vorlesungen**
Studiengang → Terminalwort

Document-Type Definitionen (DTD's)

Eine **Document-Type Definition (DTD)** ist ein Regelsystem zur Erzeugung einer Menge von XML-Dokumenten, also eine Grammatik zum Aufbau von Terminalwörtern **zusammen mit zugehörigen Ableitungsbäumen**.

Beispiel 6.40

DTD dargestellt als verallgemeinerte kontextfreie Grammatik.

Studienprofil \rightarrow **Studiengang** · **Fachsemester** · **Vorlesungen**

Studiengang \rightarrow Terminalwort

Fachsemester \rightarrow Terminalwort

Document-Type Definitionen (DTD's)

Eine **Document-Type Definition (DTD)** ist ein Regelsystem zur Erzeugung einer Menge von XML-Dokumenten, also eine Grammatik zum Aufbau von Terminalwörtern **zusammen mit zugehörigen Ableitungsbäumen**.

Beispiel 6.40

DTD dargestellt als verallgemeinerte kontextfreie Grammatik.

Studienprofil \rightarrow **Studiengang** · **Fachsemester** · **Vorlesungen**

Studiengang \rightarrow Terminalwort

Fachsemester \rightarrow Terminalwort

Vorlesungen \rightarrow **Titel***

Document-Type Definitionen (DTD's)

Eine **Document-Type Definition (DTD)** ist ein Regelsystem zur Erzeugung einer Menge von XML-Dokumenten, also eine Grammatik zum Aufbau von Terminalwörtern **zusammen mit zugehörigen Ableitungsbäumen**.

Beispiel 6.40

DTD dargestellt als verallgemeinerte kontextfreie Grammatik.

Studienprofil \rightarrow **Studiengang** · **Fachsemester** · **Vorlesungen**

Studiengang \rightarrow Terminalwort

Fachsemester \rightarrow Terminalwort

Vorlesungen \rightarrow **Titel***

Titel \rightarrow Terminalwort

Beispiel 6.40 (Forts.)

DTD dargestellt als verallgemeinerte kontextfreie Grammatik

Studienprofil \rightarrow Studiengang \cdot Fachsemester \cdot Vorlesungen

Studiengang \rightarrow Terminalwort

Fachsemester \rightarrow Terminalwort

Vorlesungen \rightarrow Titel^{*}

Titel \rightarrow Terminalwort

Beispiel 6.40 (Forts.)

DTD dargestellt als verallgemeinerte kontextfreie Grammatik

Studienprofil \rightarrow Studiengang \cdot Fachsemester \cdot Vorlesungen

Studiengang \rightarrow Terminalwort

Fachsemester \rightarrow Terminalwort

Vorlesungen \rightarrow Titel^{*}

Titel \rightarrow Terminalwort

Darstellung in Standardsyntax

```
<!DOCTYPE Studienprofil [  
  <!ELEMENT Stpr (Studgang, Fachsem, Vorlesg)>  
  <!ELEMENT Studgang (# PCDATA)>  
  <!ELEMENT Fachsem (# PCDATA)>  
  <!ELEMENT Vorlesg (Titel*)>  
  <!ELEMENT Titel (# PCDATA)>  
>
```

Beispiel 6.40 (Forts.)

DTD dargestellt als verallgemeinerte kontextfreie Grammatik

Studienprofil \rightarrow Studiengang \cdot Fachsemester \cdot Vorlesungen

Studiengang \rightarrow Terminalwort

Fachsemester \rightarrow Terminalwort

Vorlesungen \rightarrow Titel^{*}

Titel \rightarrow Terminalwort

Darstellung in Standardsyntax

```
<!DOCTYPE Studienprofil [  
  <!ELEMENT Stpr (Studgang, Fachsem, Vorlesg)>  
  <!ELEMENT Studgang (# PCDATA)>  
  <!ELEMENT Fachsem (# PCDATA)>  
  <!ELEMENT Vorlesg (Titel*)>  
  <!ELEMENT Titel (# PCDATA)>  

```

Hierbei steht # PCDATA für “parsed character data”, das heißt, für Wörter über dem Terminalalphabet.

Abschnitt 6.4

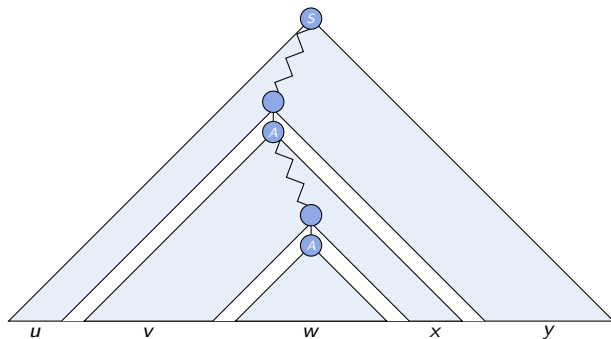
Das Pumping-Lemma für kontextfreie Sprachen

Sei \mathcal{G} eine kontextfreie Grammatik.

Ist $z \in L(\mathcal{G})$ hinreichend lang, so hat z einen Ableitungsbaum so großer Höhe, dass sich auf einem langen Ast ein Nichtterminalsymbol wiederholt.

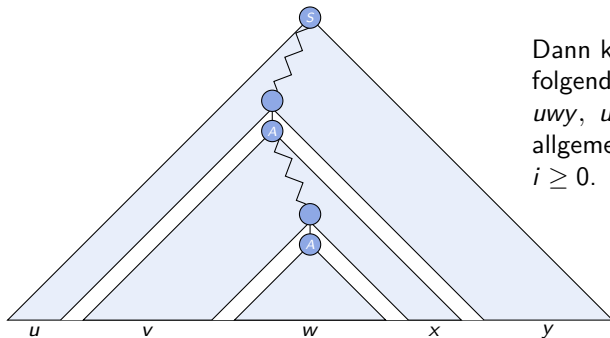
Sei \mathcal{G} eine kontextfreie Grammatik.

Ist $z \in L(\mathcal{G})$ hinreichend lang, so hat z einen Ableitungsbaum so großer Höhe, dass sich auf einem langen Ast ein Nichtterminalsymbol wiederholt.

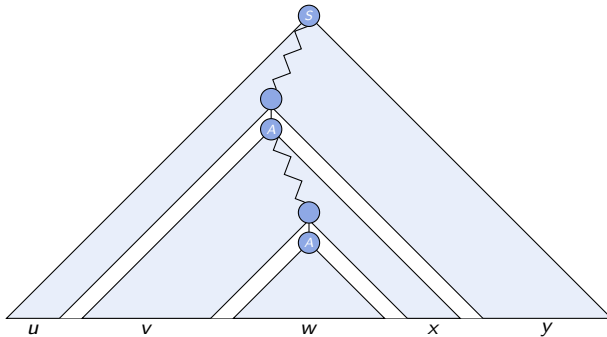


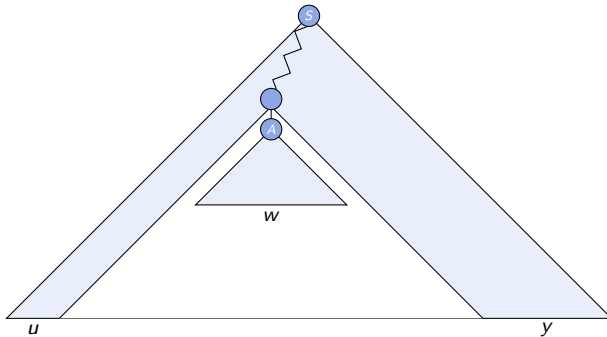
Sei \mathcal{G} eine kontextfreie Grammatik.

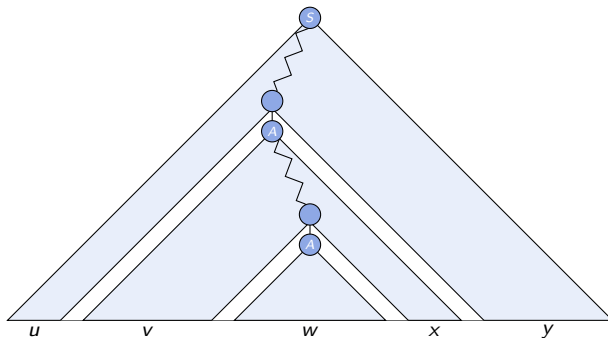
Ist $z \in L(\mathcal{G})$ hinreichend lang, so hat z einen Ableitungsbaum so großer Höhe, dass sich auf einem langen Ast ein Nichtterminalsymbol wiederholt.

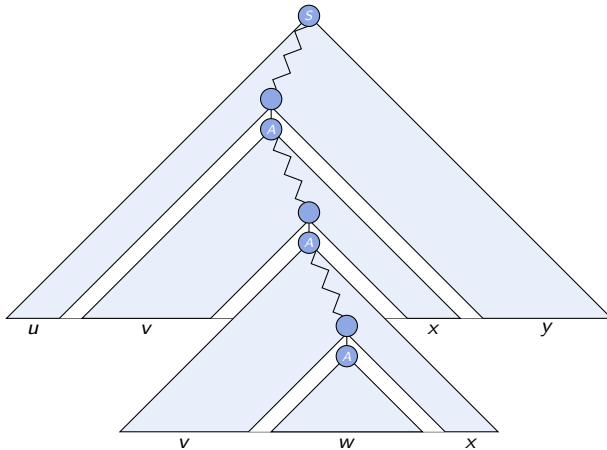


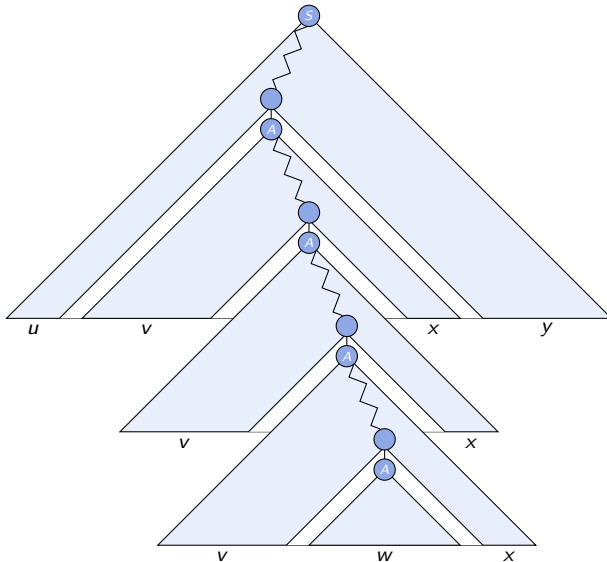
Dann kann man in \mathcal{G} auch folgende Wörter ableiten:
 $uw y$, $uvvwxy$ und
allgemein uv^iwx^iy für
 $i \geq 0$.











Lemma 6.41 (Pumping-Lemma für kontextfreie Sprachen)

Sei $L \subseteq \Sigma^*$ kontextfrei. Dann gibt es eine Zahl $n \geq 1$, so dass jedes Wort $z \in L$ mit $|z| \geq n$ zerlegbar ist in Wörter $u, v, w, x, y \in \Sigma^*$ mit

$$z = uvwxy,$$

die folgende Eigenschaften haben:

- (i) $vx \neq \varepsilon$,
- (ii) $|vwx| \leq n$,
- (iii) $uv^kwx^ky \in L$ für alle $k \geq 0$.

Beweis des Pumping-Lemmas

Lemma 6.42

Ein höchstens d -verzweigter Baum der Höhe h hat höchstens d^h Blätter.

Beweis.

Induktion über h .

Induktionsanfang $h = 0$:

Ein Baum der Höhe 0 hat $1 = d^0$ Blatt.

Induktionsschritt $h \rightarrow h + 1$:

Sei T ein Baum der Höhe $h + 1$. T lässt sich an der Wurzel in Bäume T_1, \dots, T_ℓ der Höhe $\leq h$ für ein $\ell \leq d$ zerlegen.

Induktionsannahme \implies (Zahl der Blätter von T_i) $\leq d^h$
 \implies (Zahl der Blätter von T) $\leq \ell \cdot d^h \leq d^{h+1}$. □

Beweis des Pumping-Lemmas.

Sei $\mathcal{G} = (N, \Sigma, P, S)$ eine kontextfreie Grammatik in CNF für L (bzw. $L \setminus \{\varepsilon\}$).

Wir setzen

$$n := 2^{|N|+1}.$$

Sei $z \in L$ mit $|z| > n$ und T ein Ableitungsbaum für z in G .

T ist Baum mit Verzweigungsgrad höchstens 2 (wegen CNF) mit

$|z| \geq n > 2^{|N|}$ Blättern. Nach Lemma 6.42 hat T also Höhe

$$h \geq |N| + 1.$$

Sei $p_0 = \varepsilon, p_1, p_2, \dots, p_h$ ein Ast von T der Länge h . Dabei sind die Knoten p_0, \dots, p_{h-1} innere Knoten des Baumes und deswegen mit Nichtterminalsymbolen beschriftet.

Weil $h > |N|$, gibt es i, j , so dass

$$h - 1 - |N| \leq i < j \leq h - 1$$

und $T(p_i) = T(p_j) \in N$.

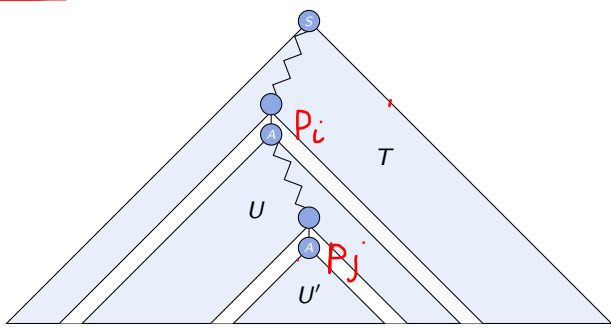
Sei U der Teilbaum von T mit Wurzel p_i .

Formal:

$$\text{dom}(U) := \{q \in \{1, 2\}^* \mid p_i q \in \text{dom}(T)\}$$

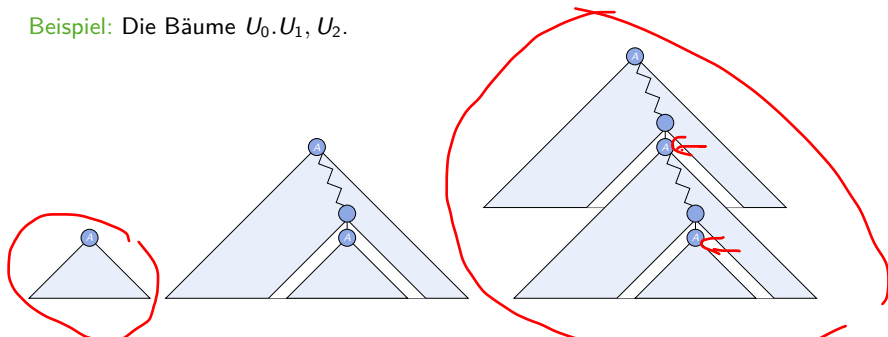
und $U(q) := T(p_i q)$.

Ähnlich sei U' der Teilbaum von T mit Wurzel p_j .



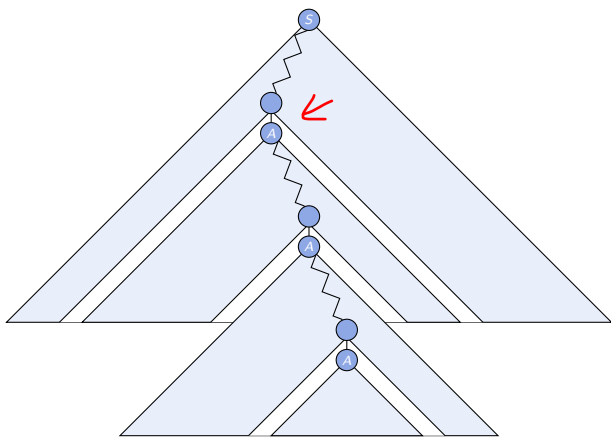
Sei $U_0 := U'$, $U_1 := U$, und für $i \geq 1$ sei U_{i+1} der Baum, der aus U entsteht, wenn man U' durch U_i ersetzt.

Beispiel: Die Bäume U_0, U_1, U_2 .



Für $i \geq 0$ sei T_i der Baum, der aus T entsteht, indem man den Teilbaum U durch U_i ersetzt.

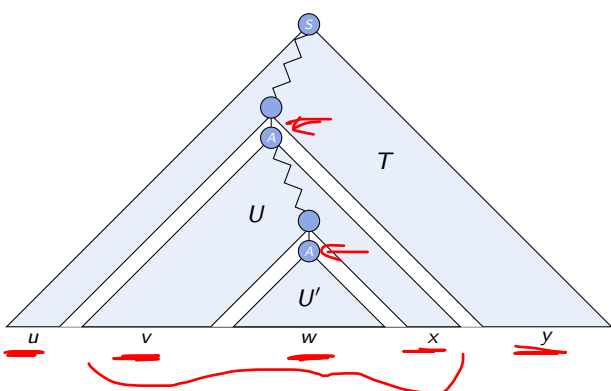
Beispiel: Der Baum T_2 .



Dann ist T_i Ableitungsbaum zu \mathcal{G} .

Sei $w := \beta(U')$. Wähle v, x so, dass $\beta(U) = vwx$.

Wähle u, y so, dass $z = \beta(T) = uvwxy$.



- ▶ Weil $i < j$ und damit $U \neq U'$ gilt $vx \neq \varepsilon$.
- ▶ Weil $i \geq h - 1 - |N|$ ist die Höhe von U höchstens $|N| + 1$. Deswegen gilt $|vwx| = |\beta(U)| \leq 2^{|N|+1} = n$.
- ▶ Für alle $k \geq 0$ gilt $uv^kwx^ky = \beta(T_k) \in L$.

□

Die Sprache $L = \{a^n b^n c^n \mid n \geq 0\} \subseteq \{a, b, c\}^*$ ist nicht kontextfrei.

Beweis.

Angenommen, L ist kontextfrei.

Wähle n zu L gemäß Pumping-Lemma und betrachte $z = a^n b^n c^n \in L$.

Pumping-Lemma liefert Zerlegung

$$z = uvwxy$$

mit $vx \neq \varepsilon$ und $|vwx| \leq n$ und $uwy \in L$.

Weil $|vwx| \leq n$ liegt das Infix vwx von $z = a^n b^n c^n$ ganz im Präfix $a^n b^n$ oder ganz im Suffix $b^n c^n$.

Fall 1: vwx liegt ganz im Präfix $a^n b^n$.

Dann sind alle c s aus z in y . Weil $vx \neq \varepsilon$ hat das Wort uwy mindestens ein a oder ein b weniger als z , aber genauso viele c s.

Also gilt $uwy \notin L$. **Widerspruch!**

Fall 2: vwx liegt ganz im Suffix $b^n c^n$.


Dann sind alle a s aus z in u . Weil $vx \neq \varepsilon$ hat das Wort uwy mindestens ein b oder ein c weniger als z , aber genauso viele a s.

Also gilt $uwy \notin L$. **Widerspruch!**



Die Sprache $L_W = \{ww \mid w \in \{a, b\}^*\}$ aller „Wiederholungswörter“ über dem Alphabet $\{a, b\}$ ist nicht kontextfrei.

Beweis.


Angenommen, L_W ist kontextfrei. 


Wähle n zu L_W gemäß Pumping-Lemma und betrachte $z = a^n b^n a^n b^n \in L$.

Pumping-Lemma liefert Zerlegung

$$z = uvwxy$$

mit $vx \neq \varepsilon$ und $|vwx| \leq n$ und $uwy \in L$.

 Weil $|vwx| \leq n$ liegt das Infix vwx von $z = a^n b^n a^n b^n$ ganz im Präfix $a^n b^n$ oder ganz im Infix $b^n a^n$ oder ganz im Suffix $a^n b^n$.


 Fall 1: vwx liegt ganz im Präfix $a^n b^n$.


Dann gilt $uwy = a^\ell b^m a^n b^n$, wobei $\ell + m < 2n$. Also gilt $uwy \notin L$. Widerspruch!

Fall 2: vwx liegt ganz im Suffix $a^n b^n$.



Analog zu Fall 1.

 Fall 3: vwx liegt ganz im Infix $b^n a^n$, aber weder im Präfix $a^n b^n$ noch im Suffix $a^n b^n$.

Dann gilt $uwy = a^n b^\ell a^m b^n$ für $1 \leq \ell, m < n$. Also $uwy \notin L$. Widerspruch! 

Abschnitt 6.5

Abschlusseigenschaften kontextfreier Sprachen

Satz 6.45

Die Klasse der kontextfreien Sprachen ist unter Vereinigung, Verkettung und Iteration abgeschlossen.

Vereinigung: Übung.

Verkettung: Seien $L_1, L_2 \subseteq \Sigma^*$ kontextfreie Sprachen. Wir wollen zeigen, dass $L_1 L_2$ kontextfrei ist.

Seien dazu $\mathcal{G}_1 = (N_1, \Sigma, P_1, S_1)$ und $\mathcal{G}_2 = (N_2, \Sigma, P_2, S_2)$ kontextfreie Grammatiken mit $L(\mathcal{G}_1) = L_1$ und $L(\mathcal{G}_2) = L_2$. OBdA gelte $N_1 \cap N_2 = \emptyset$.

Wir definieren eine Grammatik $\mathcal{G} = (N, \Sigma, P, S)$ wie folgt:

- ▶ $S \notin N_1 \cup N_2$ ist ein neues Nichtterminalsymbol;
- ▶ $N := N_1 \cup N_2 \cup \{S\}$;
- ▶ $P := P_1 \cup P_2 \cup \{S \rightarrow S_1 S_2\}$.

Behauptung: $L(\mathcal{G}) = L_1 L_2$.

Beweis.

„ \subseteq “: Sei $w \in L(\mathcal{G})$. Dann gilt $S \xrightarrow{*}_{\mathcal{G}} w$. Weil $S \rightarrow S_1 S_2$ die einzige Regel für S ist, gilt

$$S \rightarrow_{\mathcal{G}} S_1 S_2 \xrightarrow{*}_{\mathcal{G}} w.$$

Nach dem Zerlegungslemma 6.10 gibt es dann $w_1, w_2 \in \Sigma^*$, so dass $w = w_1 w_2$ mit

$$S_1 \xrightarrow{*}_{\mathcal{G}} w_1 \quad \text{und} \quad S_2 \xrightarrow{*}_{\mathcal{G}} w_2.$$

Dann gilt auch

$$S_1 \xrightarrow{*}_{\mathcal{G}_1} w_1 \quad \text{und} \quad S_2 \xrightarrow{*}_{\mathcal{G}_2} w_2.$$

(Man kann per Induktion über die Länge der Ableitung von w_i aus S_i in \mathcal{G} beweisen, dass es sich schon um eine Ableitung in \mathcal{G}_i handeln muss.)

Also $w_1 \in L(\mathcal{G}_1) = L_1$ und $w_2 \in L(\mathcal{G}_2) = L_2$ und damit $w = w_1 w_2 \in L_1 L_2$.

„ \supseteq “: Sei $w \in L_1 L_2$, etwa $w = w_1 w_2$ mit $w_1 \in L_1$ und $w_2 \in L_2$. Dann gilt

$$S_1 \xrightarrow{*}_{\mathcal{G}_1} w_1 \quad \text{und} \quad S_2 \xrightarrow{*}_{\mathcal{G}_2} w_2.$$

und damit auch

$$S_1 \xrightarrow{*}_{\mathcal{G}} w_1 \quad \text{und} \quad S_2 \xrightarrow{*}_{\mathcal{G}} w_2,$$

weil $P_1, P_2 \subseteq P$. Weil $S \rightarrow_{\mathcal{G}} S_1 S_2$ gilt nach dem Kombinationslemma 6.9

$$S \rightarrow_{\mathcal{G}} S_1 S_2 \xrightarrow{*}_{\mathcal{G}} w_1 w_2 = w.$$

Also $w \in L(\mathcal{G})$.

Iteration: Übung.

Beispiel 6.46

Betrachte die Sprachen

$$L_1 = \{a^m b^m c^n \mid m, n \geq 0\},$$

$$L_2 = \{a^m b^n c^n \mid m, n \geq 0\}.$$

Beispiel 6.46

Betrachte die Sprachen

$$L_1 = \{a^m b^m c^n \mid m, n \geq 0\},$$

$$L_2 = \{a^m b^n c^n \mid m, n \geq 0\}.$$

Beide Sprachen sind kontextfrei:

- L_1 wird erzeugt von der kontextfreien Grammatik

$$S \rightarrow AC, \quad A \rightarrow aAb \mid \varepsilon, \quad C \rightarrow cC \mid \varepsilon.$$

Beispiel 6.46

Betrachte die Sprachen

$$L_1 = \{a^m b^m c^n \mid m, n \geq 0\},$$

$$L_2 = \{a^m b^n c^n \mid m, n \geq 0\}.$$

Beide Sprachen sind kontextfrei:

- L_1 wird erzeugt von der kontextfreien Grammatik

$$S \rightarrow AC, \quad A \rightarrow aAb \mid \varepsilon, \quad C \rightarrow cC \mid \varepsilon.$$

- L_2 wird erzeugt von der kontextfreien Grammatik

$$S \rightarrow AB, \quad A \rightarrow aA \mid \varepsilon, \quad B \rightarrow bBc \mid \varepsilon.$$

Beispiel 6.46

Betrachte die Sprachen

$$L_1 = \{a^m b^m c^n \mid m, n \geq 0\},$$

$$L_2 = \{a^m b^n c^n \mid m, n \geq 0\}.$$

Beide Sprachen sind kontextfrei:

- L_1 wird erzeugt von der kontextfreien Grammatik

$$S \rightarrow AC, \quad A \rightarrow aAb \mid \varepsilon, \quad C \rightarrow cC \mid \varepsilon.$$



- L_2 wird erzeugt von der kontextfreien Grammatik

$$S \rightarrow AB, \quad A \rightarrow aA \mid \varepsilon, \quad B \rightarrow bBc \mid \varepsilon.$$

Aber es gilt

$$L_1 \cap L_2 = \{a^n b^n c^n \mid n \in \mathbb{N}\},$$



und nach Beispiel 6.43 ist diese Sprache nicht kontextfrei.

Bemerkung 6.47

Weil die Klasse der kontextfreien Sprachen abgeschlossen ist unter Vereinigung, aber nicht unter Durchschnitt, kann sie nicht unter Komplement abgeschlossen sein, denn Durchschnitt lässt sich mit Hilfe der de Morgan'schen Regel aus Vereinigung und Komplement darstellen.

Bemerkung 6.47

Weil die Klasse der kontextfreien Sprachen abgeschlossen ist unter Vereinigung, aber nicht unter Durchschnitt, kann sie nicht unter Komplement abgeschlossen sein, denn Durchschnitt lässt sich mit Hilfe der de Morgan'schen Regel aus Vereinigung und Komplement darstellen.

Beispiel 6.48

Nach Beispiel 6.44 ist die Sprache $L_W = \{ww \mid w \in \{a, b\}^*\}$ nicht kontextfrei.

Bemerkung 6.47

Weil die Klasse der kontextfreien Sprachen abgeschlossen ist unter Vereinigung, aber nicht unter Durchschnitt, kann sie nicht unter Komplement abgeschlossen sein, denn Durchschnitt lässt sich mit Hilfe der de Morgan'schen Regel aus Vereinigung und Komplement darstellen.

Beispiel 6.48

Nach Beispiel 6.44 ist die Sprache $L_W = \{ww \mid w \in \{a, b\}^*\}$ nicht kontextfrei.

Folgende kontextfreie Grammatik erzeugt $\overline{L_W}$ (Beweis als Übung).

$$\begin{aligned}
 &\rightarrow S \rightarrow AB \mid BA \mid \boxed{A \mid B} \\
 &\rightarrow A \rightarrow CAC \mid a \\
 &\rightarrow B \rightarrow CBC \mid b \\
 &\quad C \rightarrow a \mid b.
 \end{aligned}$$

Also ist $\overline{L_W}$ kontextfrei, aber $\overline{\overline{L_W}} = L_W$ nicht.