Einführung in die Computerlinguistik – Formale Grammatiken rechtslineare und kontextfreie Grammatiken Kellerautomaten

Dozentin: Wiebke Petersen

13. Foliensatz

Formale Grammatik

Definition

Eine formale Grammatik *ist ein 4-Tupel* $G = (N, T, S, P)$ *aus*

- *einem Alphabet von Terminalsymbolen* T *(häufig auch* Σ*)*
- *einem Alphabet von Nichtterminalsymbolen* N *mit* $N \cap T = \emptyset$
- *einem Startsymbol* $S \in N$
- *einer Menge von Regeln/Produktionen*
  $P \subseteq \{\langle \alpha, \beta \rangle \mid \alpha, \beta \in (N \cup T)^* \text{ und } \alpha \notin T^*\}$.

Für eine Regel $\langle \alpha, \beta \rangle$ *schreiben wir auch* $\alpha \to \beta$.

S	→	NP VP	VP	→	V	NP	→	D N
D	→	the	N	→	cat	V	→	sleeps

Generiert: the cat sleeps

Konvention: Wir verwenden Großbuchstaben für Nichtterminalsymbole und Kleinbuchstaben für Terminalsymbole

Terminologie

$$G = \langle\{\mathrm{S,NP,VP,V,D,N,EN}\}, \{\mathrm{the,\ cat,\ peter,\ chases}\}, \mathrm{S}, P\rangle$$

$$P = \left\{ \begin{array}{rcl rcl rcl} \mathrm{S} & \rightarrow & \mathrm{NP\ VP} & \mathrm{VP} & \rightarrow & \mathrm{V\ NP} & \mathrm{NP} & \rightarrow & \mathrm{D\ N} \\ \mathrm{NP} & \rightarrow & \mathrm{EN} & \mathrm{D} & \rightarrow & \mathrm{the} & \mathrm{N} & \rightarrow & \mathrm{cat} \\ \mathrm{EN} & \rightarrow & \mathrm{peter} & \mathrm{V} & \rightarrow & \mathrm{chases} & & & \end{array} \right\}$$

Terminologie

$$G = \langle \{\mathrm{S,NP,VP,V,D,N,EN}\}, \{\mathrm{the,\ cat,\ peter,\ chases}\}, \mathrm{S}, P \rangle$$

$$P = \left\{ \begin{array}{rcl rcl rcl} \mathrm{S} & \rightarrow & \mathrm{NP\ VP} & \mathrm{VP} & \rightarrow & \mathrm{V\ NP} & \mathrm{NP} & \rightarrow & \mathrm{D\ N} \\ \mathrm{NP} & \rightarrow & \mathrm{EN} & \mathrm{D} & \rightarrow & \mathrm{the} & \mathrm{N} & \rightarrow & \mathrm{cat} \\ \mathrm{EN} & \rightarrow & \mathrm{peter} & \mathrm{V} & \rightarrow & \mathrm{chases} & & & \end{array} \right\}$$

“NP VP” ist in einem Schritt aus S ableitbar

Terminologie

$G = \langle\{\text{S,NP,VP,V,D,N,EN}\}, \{\text{the, cat, peter, chases}\}, \text{S}, P\rangle$

$$P = \left\{ \begin{array}{rcl rcl rcl} \text{S} & \rightarrow & \text{NP VP} & \text{VP} & \rightarrow & \text{V NP} & \text{NP} & \rightarrow & \text{D N} \\ \text{NP} & \rightarrow & \text{EN} & \text{D} & \rightarrow & \text{the} & \text{N} & \rightarrow & \text{cat} \\ \text{EN} & \rightarrow & \text{peter} & \text{V} & \rightarrow & \text{chases} & & & \end{array} \right\}$$

"NP VP" ist in einem Schritt aus S ableitbar

"the cat chases peter" ist ableitbar aus S:
Ableitung:

$$\begin{array}{llll} \text{S} & \rightarrow \text{NP VP} & \rightarrow \text{NP V NP} & \rightarrow \text{NP V EN} \\ & \rightarrow \text{NP V peter} & \rightarrow \text{NP chases peter} & \rightarrow \text{D N chases peter} \\ & \rightarrow \text{D cat chases peter} & \rightarrow \text{the cat chases peter} & \end{array}$$

Terminologie

$G = \langle \{S, NP, VP, V, D, N, EN\}, \{\text{the, cat, peter, chases}\}, S, P \rangle$

$$P = \left\{ \begin{array}{rcl rcl rcl} S & \to & NP\ VP & VP & \to & V\ NP & NP & \to & D\ N \\ NP & \to & EN & D & \to & \text{the} & N & \to & \text{cat} \\ EN & \to & \text{peter} & V & \to & \text{chases} & & & \end{array} \right\}$$

"NP VP" ist in einem Schritt aus S ableitbar

"the cat chases peter" ist ableitbar aus S:
Ableitung:

$$\begin{array}{lll} S \to NP\ VP & \to NP\ V\ NP & \to NP\ V\ EN \\ \to NP\ V\ \text{peter} & \to NP\ \text{chases peter} & \to D\ N\ \text{chases peter} \\ \to D\ \text{cat chases peter} & \to \text{the cat chases peter} & \end{array}$$

Die Menge aller aus dem Startsymbol S ableitbarer Wörter ist die von der Grammatik G erzeugte Sprache $L(G)$.

$$L(G) = \left\{ \begin{array}{cc} \text{the cat chases peter,} & \text{peter chases the cat,} \\ \text{peter chases peter,} & \text{the cat chases the cat} \end{array} \right\}$$

rechtslineare Grammatiken

Definition

Eine Grammatik (N, T, S, P) *heißt rechtslinear, wenn alle Regeln/Produktionen die folgende Form haben:*
$A \to a$ *oder* $A \to aB$ *wobei* $a \in T \cup \{\varepsilon\}$ *und* $A, B \in N$.
Eine durch eine rechtslineare Grammatik erzeugte Sprache heißt rechtslinear.

(Vorsicht, häufig werden schärfere [aber äquivalente] Bedingungen gefordert!)

rechtslineare Grammatiken

Definition

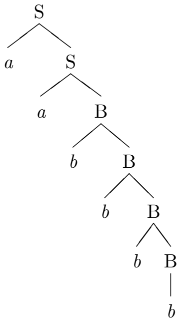
Eine Grammatik (N, T, S, P) heißt rechtslinear, wenn alle Regeln/Produktionen die folgende Form haben:
$A \rightarrow a$ *oder* $A \rightarrow aB$ *wobei* $a \in T \cup \{\varepsilon\}$ *und* $A, B \in N$.
Eine durch eine rechtslineare Grammatik erzeugte Sprache heißt rechtslinear.

(Vorsicht, häufig werden schärfere [aber äquivalente] Bedingungen gefordert!)

Beispiel:
$G = (\{S, B\}, \{a, b\}, S, P)$ mit

$$P = \left\{ \begin{array}{lcl} \mathrm{S} & \rightarrow & \mathrm{a\ S} \\ \mathrm{S} & \rightarrow & \mathrm{a\ B} \\ \mathrm{B} & \rightarrow & \mathrm{b\ B} \\ \mathrm{B} & \rightarrow & \mathrm{b} \end{array} \right\}$$

G generiert $L(G) = L(a^+b^+)$



Ableitungsbaum ($aabbbb \in L(G)$)

rechtslineare Grammatiken und reguläre Sprachen

Theorem

Sei L eine formale Sprache, dann sind die folgenden Aussagen äquivalent:

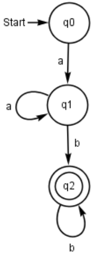
1. *L ist regulär.*
2. *Es gibt eine rechtslineare Grammatik G, die L erzeugt.*
3. *Es gibt einen endlichen Automaten A, der L akzteptiert.*
4. *Es gibt einen regulären Ausdruck R, der L beschreibt.*

rechtslineare Grammatik:

$G = (\{S, B\}, \{a, b\}, S, P)$ mit

$$P = \left\{ \begin{array}{ccc} \mathrm{S} & \rightarrow & \mathrm{a\ S} \\ \mathrm{S} & \rightarrow & \mathrm{a\ B} \\ \mathrm{B} & \rightarrow & \mathrm{b\ B} \\ \mathrm{B} & \rightarrow & \mathrm{b} \end{array} \right\}$$

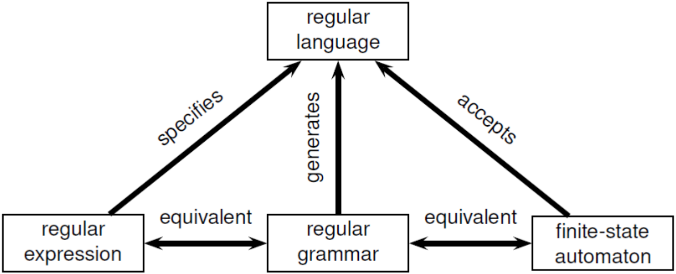
endlicher Automat:



regulärer Ausdruck:

$a^+ b^+$

Zusammenfassung: reguläre Sprachen



kontextfreie Grammatik

Definition

Eine Grammatik (N, T, S, P) heißt kontextfrei, wenn alle Regeln/Produktionen die folgende Form haben:

$$A \rightarrow \alpha, \text{ wobei } A \in N \text{ und } \alpha \in (T \cup N)^*.$$

Eine durch eine kontextfreie Grammatik erzeugte Sprache heißt kontextfrei.

kontextfreie Grammatik

Definition

Eine Grammatik (N, T, S, P) heißt kontextfrei, wenn alle Regeln/Produktionen die folgende Form haben:

$$A \to \alpha, \text{ wobei } A \in N \text{ und } \alpha \in (T \cup N)^*.$$

Eine durch eine kontextfreie Grammatik erzeugte Sprache heißt kontextfrei.

Die Menge der kontextfreien Sprachen ist eine echte Obermenge der Menge der regulären Sprachen

kontextfreie Grammatik

Definition

Eine Grammatik (N, T, S, P) heißt kontextfrei, wenn alle Regeln/Produktionen die folgende Form haben:

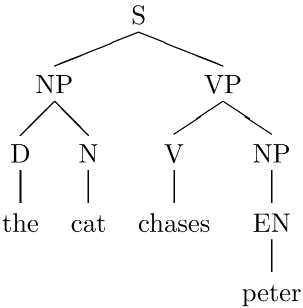
$$A \to \alpha, \text{ wobei } A \in N \text{ und } \alpha \in (T \cup N)^*.$$

Eine durch eine kontextfreie Grammatik erzeugte Sprache heißt kontextfrei.

Die Menge der kontextfreien Sprachen ist eine echte Obermenge der Menge der regulären Sprachen

Beweis: Jede reguläre Sprache ist per Definition auch kontextfrei und es gibt mindestens eine kontextfreie Sprache, nämlich $L(a^n b^n)$ mit $n \geq 0$, die nicht regulär ist. $(S \to aSb, S \to \varepsilon)$

Ableitungsbaum



Beispiel einer kontextfreien Sprache

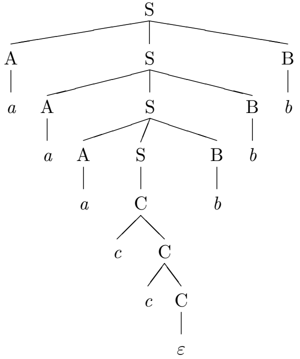
$$G = \langle \{\mathrm{S}, \mathrm{A}, \mathrm{B}, \mathrm{C}\}, \{a, b, c\}, \mathrm{S}, P \rangle$$

$$P = \left\{ \begin{array}{llllll} \mathrm{S} & \rightarrow & \mathrm{ASB} & \mathrm{S} & \rightarrow & \mathrm{C} \\ \mathrm{A} & \rightarrow & a & \mathrm{B} & \rightarrow & b \\ \mathrm{C} & \rightarrow & c\mathrm{C} & \mathrm{C} & \rightarrow & \varepsilon \end{array} \right\}$$

Beispiel einer kontextfreien Sprache

$$G = \langle \{\mathrm{S}, \mathrm{A}, \mathrm{B}, \mathrm{C}\}, \{a, b, c\}, \mathrm{S}, P \rangle$$

$$P = \left\{ \begin{array}{lllllll} \mathrm{S} & \rightarrow & \mathrm{ASB} & & \mathrm{S} & \rightarrow & \mathrm{C} \\ \mathrm{A} & \rightarrow & a & & \mathrm{B} & \rightarrow & b \\ \mathrm{C} & \rightarrow & cC & & \mathrm{C} & \rightarrow & \varepsilon \end{array} \right\}$$



Übung: Welche Sprache generiert diese Grammatik? Können Sie eine äquivalente Grammatik angeben, die mit weniger Nichtterminalsymbolen auskommt?

Linksableitung

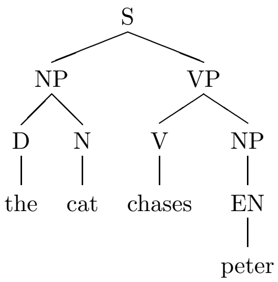
Gegeben eine kontextfreie Grammatik G. Eine Ableitung bei der stets das am weitesten links stehende nichtterminale Symbol ersetzt wird, heißt Linksableitung

S $\rightarrow$ NP VP $\rightarrow$ D N VP $\rightarrow$ the N VP
$\rightarrow$ the cat VP $\rightarrow$ the cat V NP $\rightarrow$ the cat chases NP
$\rightarrow$ the cat chases EN $\rightarrow$ the cat chases peter

Linksableitung

Gegeben eine kontextfreie Grammatik G. Eine Ableitung bei der stets das am weitesten links stehende nichtterminale Symbol ersetzt wird, heißt Linksableitung

S	→ NP VP	→ D N VP	→ the N VP
	→ the cat VP	→ the cat V NP	→ the cat chases NP
	→ the cat chases EN	→ the cat chases peter	



Zu jeder Linksableitung gibt es genau einen Ableitungsbaum und zu jedem Ableitungsbaum gibt es genau eine Linksableitung.

ambige Grammatik

Eine Grammatik G heißt ambig, wenn es für ein Wort $w \in L(G)$ mehr als eine Linksableitung gibt.

$G = (N, T, \mathrm{NP}, P)$ mit $N = \{\mathrm{S}, \mathrm{EN}, \mathrm{NP}, \mathrm{VP}, \mathrm{PP}, \mathrm{D}, \mathrm{N}, \mathrm{P}\}$,
$T = \{\mathrm{Eva}, \mathrm{sieht}, \mathrm{den}, \mathrm{Mann}, \mathrm{mit}, \mathrm{dem}, \mathrm{Fernglas}\}$,

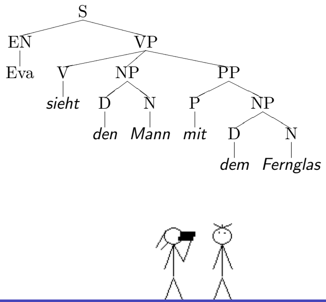
$$P = \left\{ \begin{array}{rcl rcl rcl} \mathrm{S} & \rightarrow & \mathrm{EN\ VP} & \mathrm{VP} & \rightarrow & \mathrm{V\ NP} & \mathrm{VP} & \rightarrow & \mathrm{V\ NP\ PP} \\ \mathrm{NP} & \rightarrow & \mathrm{D\ N} & \mathrm{NP} & \rightarrow & \mathrm{D\ N\ PP} & \mathrm{PP} & \rightarrow & \mathrm{P\ NP} \\ \mathrm{EN} & \rightarrow & \mathrm{Eva} & \mathrm{P} & \rightarrow & \mathrm{mit} & \mathrm{V} & \rightarrow & \mathrm{sieht} \\ \mathrm{D} & \rightarrow & \mathrm{den} & \mathrm{D} & \rightarrow & \mathrm{dem} & \mathrm{N} & \rightarrow & \mathrm{Mann} \\ \mathrm{N} & \rightarrow & \mathrm{Fernglas} & & & & & & \end{array} \right\}$$

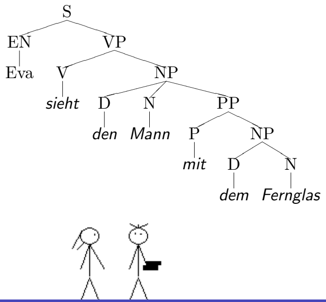
ambige Grammatik

Eine Grammatik G heißt ambig, wenn es für ein Wort $w \in L(G)$ mehr als eine Linksableitung gibt.

$G = (N, T, \mathrm{NP}, P)$ mit $N = \{\mathrm{S}, \mathrm{EN}, \mathrm{NP}, \mathrm{VP}, \mathrm{PP}, \mathrm{D}, \mathrm{N}, \mathrm{P}\}$,
$T = \{\mathrm{Eva}, \mathrm{sieht}, \mathrm{den}, \mathrm{Mann}, \mathrm{mit}, \mathrm{dem}, \mathrm{Fernglas}\}$,

$$P = \left\{ \begin{array}{rclrclrcl} \mathrm{S} & \rightarrow & \mathrm{EN\ VP} & \mathrm{VP} & \rightarrow & \mathrm{V\ NP} & \mathrm{VP} & \rightarrow & \mathrm{V\ NP\ PP} \\ \mathrm{NP} & \rightarrow & \mathrm{D\ N} & \mathrm{NP} & \rightarrow & \mathrm{D\ N\ PP} & \mathrm{PP} & \rightarrow & \mathrm{P\ NP} \\ \mathrm{EN} & \rightarrow & \mathrm{Eva} & \mathrm{P} & \rightarrow & \mathrm{mit} & \mathrm{V} & \rightarrow & \mathrm{sieht} \\ \mathrm{D} & \rightarrow & \mathrm{den} & \mathrm{D} & \rightarrow & \mathrm{dem} & \mathrm{N} & \rightarrow & \mathrm{Mann} \\ \mathrm{N} & \rightarrow & \mathrm{Fernglas} & & & & & & \end{array} \right\}$$





Kellerautomaten

Ziel: Automatenmodell mit dem genau die kontextfreien Sprachen akzeptiert werden können (analog zu endlichen Automaten und regulären Sprachen).

Kellerautomaten

Ziel: Automatenmodell mit dem genau die kontextfreien Sprachen akzeptiert werden können (analog zu endlichen Automaten und regulären Sprachen).

Lösung: Hinzunahme eines unbeschränkten Speichers in Form eines Stapels, von dessen Spitze etwas genommen und auf dessen Spitze etwas abgelegt werden kann.

Kellerautomaten entstehen aus endlichen Automaten durch

- Hinzunahme eines Kelleralphabets
- Erweiterung der Transitionen (es muss das Lesen und Ersetzen der Kellerspitze realisiert werden)

Informelles Beispiel

Wir betrachten die Sprache $\{a^i b^i \mid i > 0\}$.

Das Akzeptieren eines Eingabewortes geschieht wie folgt:

1. Aufbau des Kellers: für jedes gelesene a lege ein Symbol auf dem Keller ab (wir nehmen das Symbol Z)
2. Abbau des Kellers: für jedes gelesene b nehme ein Symbol Z vom Keller herunter
3. durch zwei Kontrollzustände sorge dafür, dass Aufbau und Abbau nur in dieser Reihenfolge möglich sind (Aufbau mit q_0, Abbau mit q_1, keine Rückkehr nach q_0)
4. Akzeptiere, wenn am Ende des Eingabewortes der Kellerboden erreicht ist.

Der Keller realisiert in diesem Fall eine Zählervariable.

Transition

$$(p, a, Z, v, q)$$

Transition

(p, a, Z, v, q)

aktueller Zustand

Transition

(p, a, Z, v, q)

aktueller Zustand

zu lesendes Eingabesymbol

Transition

oberstes Symbol auf dem Stack
(wird entfernt)
- pop up -

(p, a, Z, v, q)

aktueller Zustand

zu lesendes Eingabesymbol

Transition

oberstes Symbol auf dem Stack
(wird entfernt)
- pop up -

(p, a, Z, v, q)

aktueller Zustand

zu lesendes Eingabesymbol

Wort, das auf den Stack gelegt wird
- push down -

Transition

oberstes Symbol auf dem Stack
(wird entfernt)
- pop up -

neuer Zustand

(p, a, Z, v, q)

aktueller Zustand

zu lesendes Eingabesymbol

Wort, das auf den Stack gelegt wird
- push down -

Transition

oberstes Symbol auf dem Stack
(wird entfernt)
- pop up -

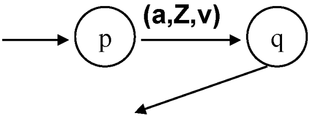
neuer Zustand

(p, a, Z, v, q)

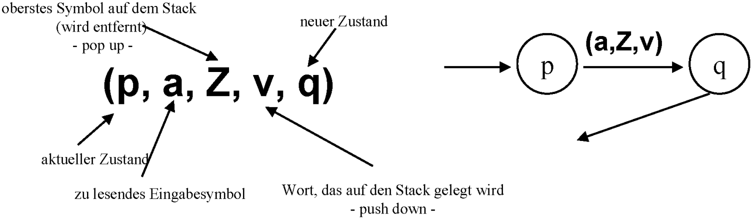
aktueller Zustand

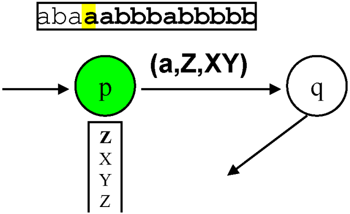
zu lesendes Eingabesymbol

Wort, das auf den Stack gelegt wird
- push down -

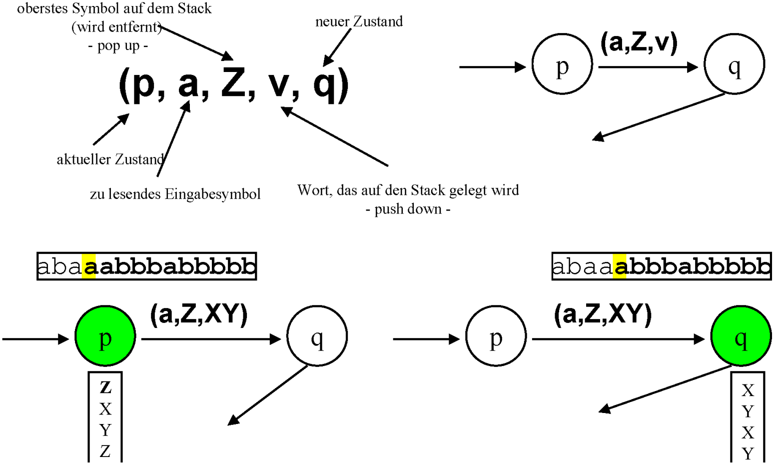


Transition

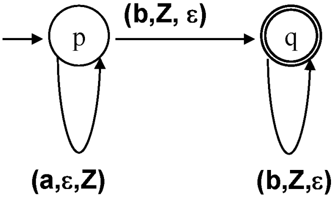

oberstes Symbol auf dem Stack
(wird entfernt)
- pop up -
neuer Zustand
(p, a, Z, v, q)
aktueller Zustand
zu lesendes Eingabesymbol
Wort, das auf den Stack gelegt wird
- push down -
p
(a,Z,v)
q


abaaabbabbbbb
p
(a,Z,XY)
q
Z
X
Y
Z

Transition

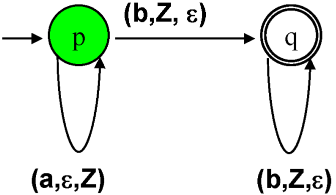

oberstes Symbol auf dem Stack
(wird entfernt)
- pop up -
neuer Zustand
(p, a, Z, v, q)
aktueller Zustand
zu lesendes Eingabesymbol
Wort, das auf den Stack gelegt wird
- push down -
p
(a,Z,v)
q
abaaabbbabbbbb
p
(a,Z,XY)
q
Z
X
Y
Z
abaaabbbabbbbb
p
(a,Z,XY)
q
X
Y
X
Y

Beispiel eines Kellerautomaten



dieser Kellerautomat akzeptiert
die Sprache $a^n b^n$

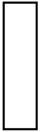
Arbeitsweise eines Kellerautomaten



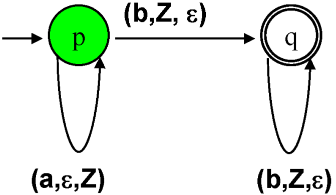
(noch) zu lesendes Wort:

a	a	a	b	b	b

aktueller Stack:

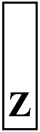


Arbeitsweise eines Kellerautomaten

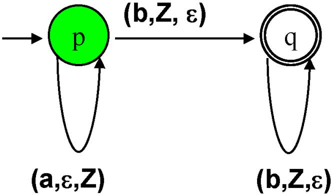


(noch) zu lesendes Wort:

aktueller Stack:



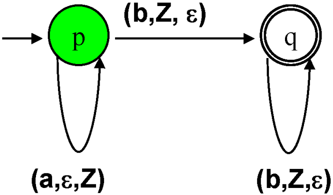
Arbeitsweise eines Kellerautomaten



(noch) zu lesendes Wort:

aktueller Stack:

Arbeitsweise eines Kellerautomaten



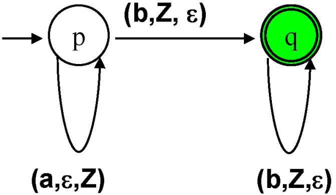
(noch) zu lesendes Wort:

b b b

aktueller Stack:

Z
Z
Z

Arbeitsweise eines Kellerautomaten



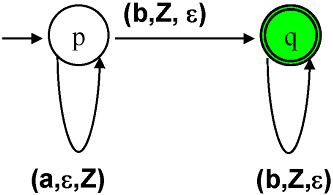
(noch) zu lesendes Wort:

b b

aktueller Stack:

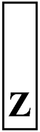
Z
Z

Arbeitsweise eines Kellerautomaten

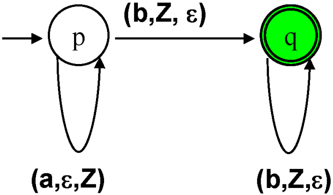


(noch) zu lesendes Wort:

aktueller Stack:

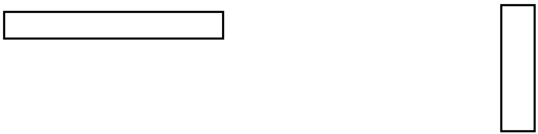


Arbeitsweise eines Kellerautomaten

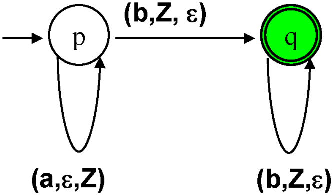


(noch) zu lesendes Wort:

aktueller Stack:



Arbeitsweise eines Kellerautomaten

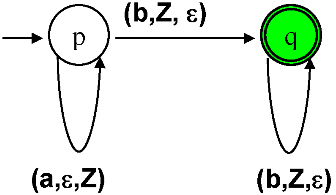


(noch) zu lesendes Wort:

aktueller Stack:

Das Wort ist abgearbeitet!

Arbeitsweise eines Kellerautomaten



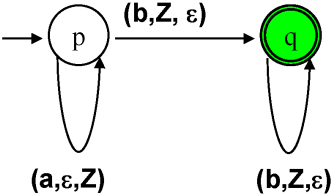
(noch) zu lesendes Wort:

Das Wort ist abgearbeitet!

aktueller Stack:

Der Stack ist leer!

Arbeitsweise eines Kellerautomaten

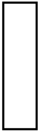


Der Automat befindet sich in einem Endzustand!

(noch) zu lesendes Wort:

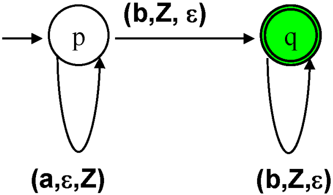
Das Wort ist abgearbeitet!

aktueller Stack:



Der Stack ist leer!

Arbeitsweise eines Kellerautomaten



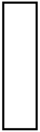
Der Automat befindet sich in einem Endzustand!

Darum akzeptiert der Kellerautomat das Wort!

(noch) zu lesendes Wort:

Das Wort ist abgearbeitet!

aktueller Stack:



Der Stack ist leer!

Von kontextfreien Grammatiken zu Kellerautomaten

- Für das Akzeptieren einer kontextfreien Sprache genügt ein Kellerautomat mit nur zwei Zuständen, wobei die einzige Aufgabe des Startzustands darin besteht, das Startsymbol S der Grammatik in den Keller zu legen.

Von kontextfreien Grammatiken zu Kellerautomaten

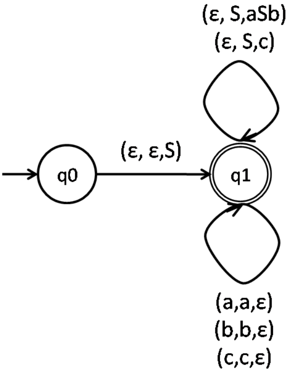
- Für das Akzeptieren einer kontextfreien Sprache genügt ein Kellerautomat mit nur zwei Zuständen, wobei die einzige Aufgabe des Startzustands darin besteht, das Startsymbol S der Grammatik in den Keller zu legen.
- Während der eigentlichen Rechnung befindet sich der Automat permanent in dem selben Zustand (dem Nichtstartzustand), die Rechnung findet nur in dem Keller statt.

Von kontextfreien Grammatiken zu Kellerautomaten

- Für das Akzeptieren einer kontextfreien Sprache genügt ein Kellerautomat mit nur zwei Zuständen, wobei die einzige Aufgabe des Startzustands darin besteht, das Startsymbol S der Grammatik in den Keller zu legen.
- Während der eigentlichen Rechnung befindet sich der Automat permanent in dem selben Zustand (dem Nichtstartzustand), die Rechnung findet nur in dem Keller statt.
- Der konstruierte Automat vollzieht zwei verschiedene Arbeitsschritte:
  - Nicht-Leseschritt mit Bezug auf Grammatikregel $B \to \beta$:
    Ersetze die Kellerspitze B mit β (Expansion).
  - Leseschritte:
    Lies ein $a \in T$ der Eingabekette und entferne a von der Kellerspitze (Scan).

Beispiel: Von kontextfreien Grammatiken zu Kellerautomaten

Grammatik: $(\{S\}, \{a, b, c\}, S, \{S \rightarrow aSb, S \rightarrow c\})$
generierte Sprache: $L(a^n cb^n)$



Hausaufgaben

1. Sei L die Sprache, die aus allen nichtleeren Wörtern über dem Alphabet $\{a, b\}$ besteht, in denen auf jedes a unmittelbar ein b folgt. Beispiele für Wörter dieser Sprache: *bbbab*, *abababab*, *bb*, *babbbbab*.
   - **(a)** geben Sie eine rechtslineare Grammatik G an, die L erzeugt und zeichnen Sie den Ableitungsbaum für das Wort *bbababb*
   - **(b)** geben Sie einen endlichen Automaten A an, der L akzteptiert.
   - **(c)** geben Sie einen regulären Ausdruck R an, der L beschreibt.
2. Geben sie jeweils eine kontextfreie Grammatik zu den folgenden Sprachen an:
   - **(a)** $L_1 = \{a^i b^j | i > j > 0\}$
   - **(b)** $L_2 = \{w \in \{a, b\}^* | w \text{ ist ein Palindrom}\}$

   Wählen Sie pro Sprache ein Wort, das mindestens die Länge 5 hat, und zeichnen Sie den Ableitungsbaum in Bezug auf Ihre Grammatik.