

# Einführung in die Computerlinguistik – Kontextfreie Grammatiken und Kellerautomaten

Dozentin: Wiebke Petersen

28.6.2010

# 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**.

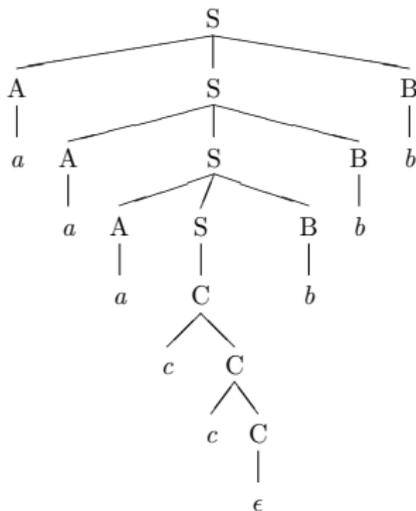
Die Menge der kontextfreien Sprachen ist eine echte Obermenge der Menge der regulären Sprachen

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

# Beispiel einer kontextfreien Sprache

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

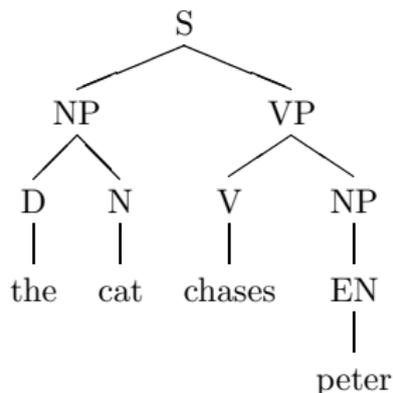
$$P = \left\{ \begin{array}{ll} S \rightarrow ASB & S \rightarrow C \\ A \rightarrow a & B \rightarrow b \\ C \rightarrow cC & C \rightarrow \epsilon \end{array} \right\}$$



# Linksableitung

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

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



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

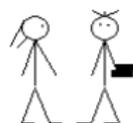
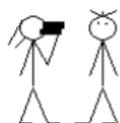
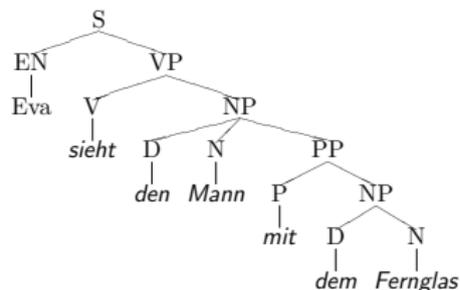
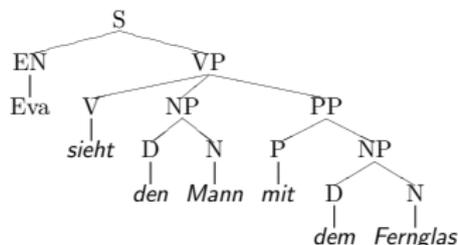
# ambige Grammatik

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

$G = (N, T, NP, P)$  mit  $N = \{S, EN, NP, VP, PP, D, N, P\}$ ,

$T = \{\text{Eva, sieht, den, Mann, mit, dem, Fernglas}\}$ ,

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



# 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

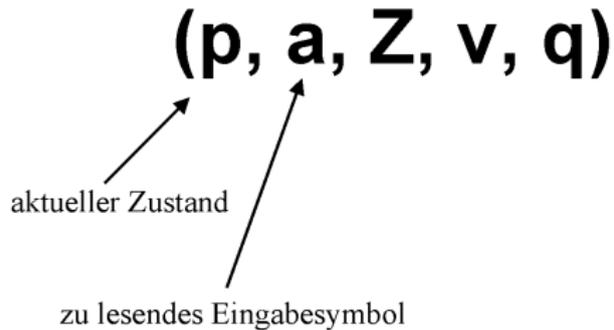
**(p, a, Z, v, q)**

# Transition

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

aktueller Zustand 

# Transition



## Transition

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

**(p, a, Z̄, v, q)**

aktueller Zustand

zu lesendes Eingabesymbol

## Transition

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

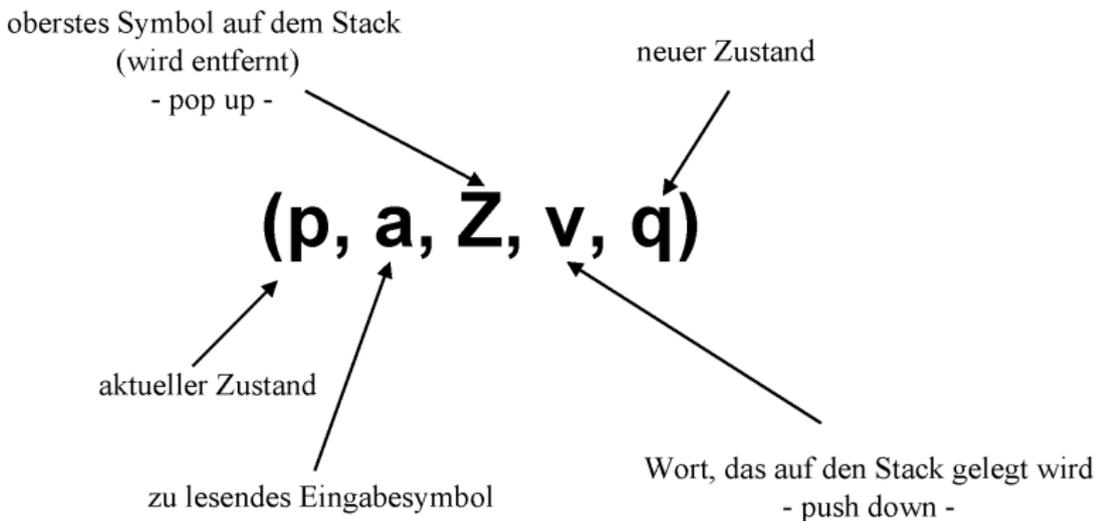
**(p, a, Z, v, q)**

aktueller Zustand

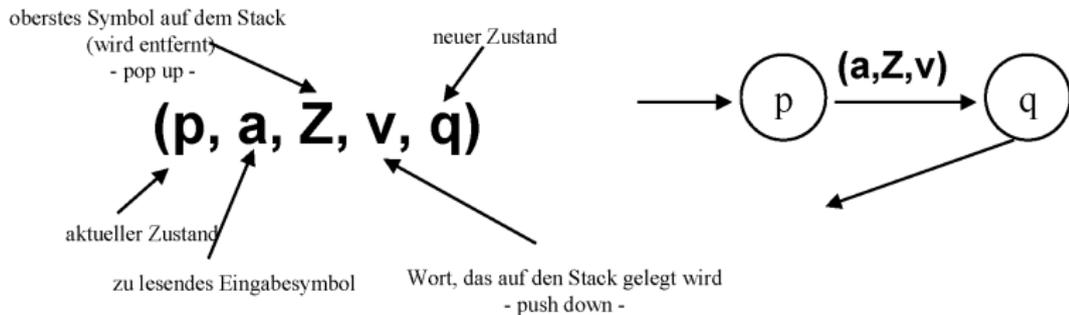
zu lesendes Eingabesymbol

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

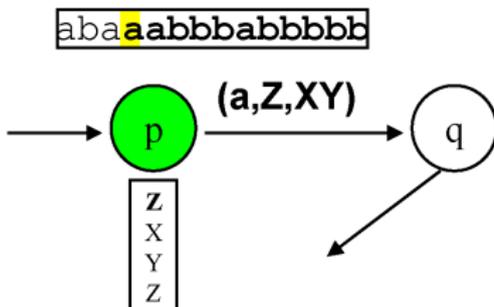
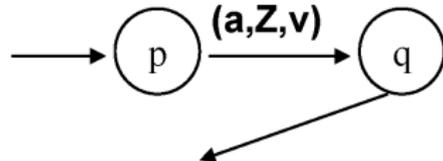
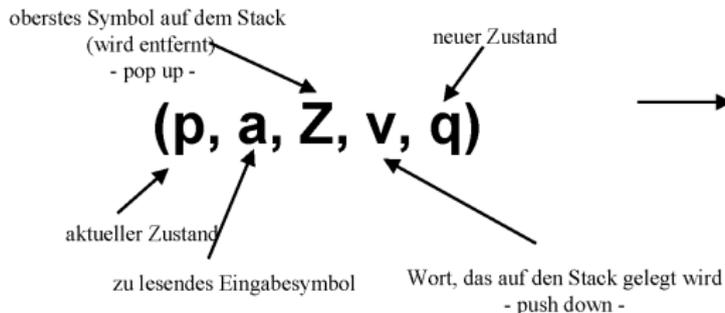
## Transition



# Transition



# Transition



# Transition

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

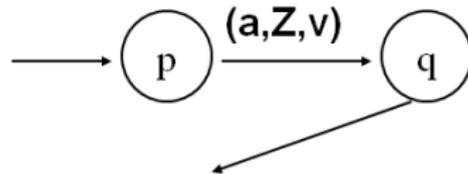
neuer Zustand

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

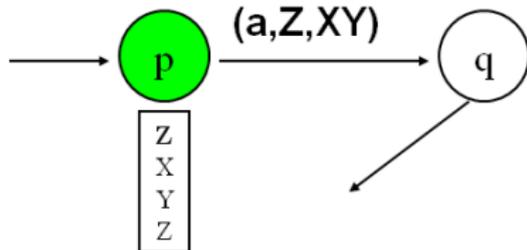
aktueller Zustand

zu lesendes Eingabesymbol

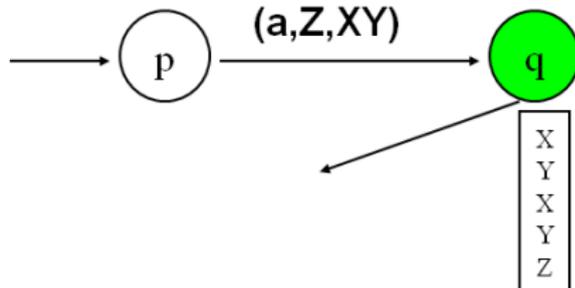
Wort, das auf den Stack gelegt wird  
- push down -



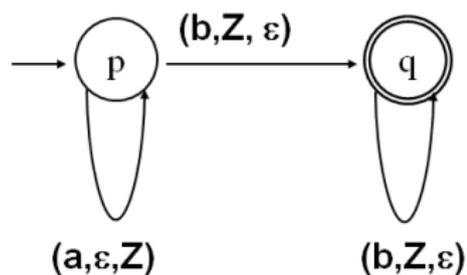
aba**a**abbbabbbb



abaa**a**abbbabbbb

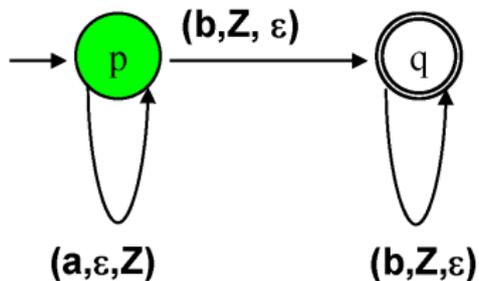


## Beispiel eines Kellerautomaten



dieser Kellerautomat akzeptiert  
die Sprache  $L(a^n b^n)$

## Arbeitsweise eines Kellerautomaten



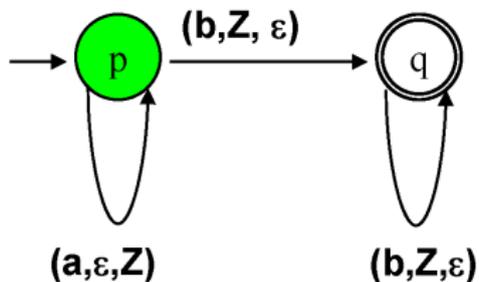
(noch) zu lesendes Wort:

a	a	a	b	b	b
---	---	---	---	---	---

aktueller Stack:



## Arbeitsweise eines Kellerautomaten



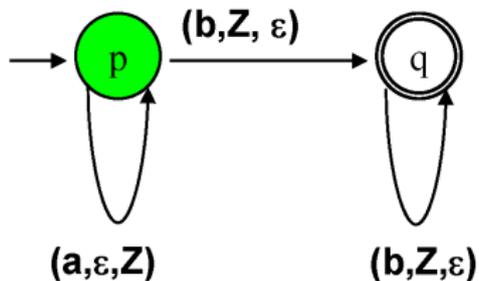
(noch) zu lesendes Wort:

a	a	b	b	b
---	---	---	---	---

aktueller Stack:

Z
---

## Arbeitsweise eines Kellerautomaten



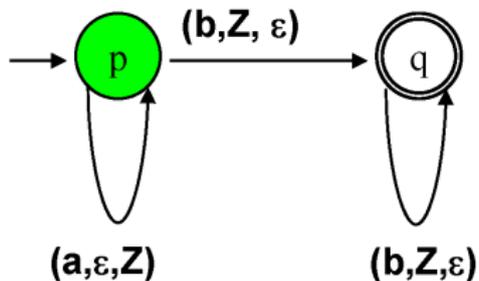
(noch) zu lesendes Wort:

a b b b
---------

aktueller Stack:

Z
Z

## Arbeitsweise eines Kellerautomaten



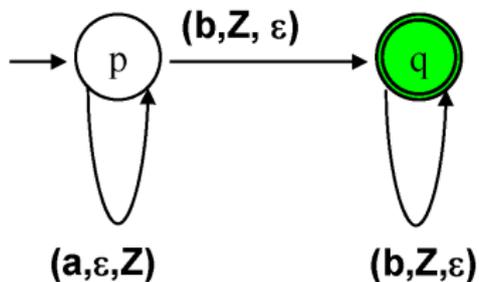
(noch) zu lesendes Wort:

b b b
-------

aktueller Stack:

Z
Z
Z

## Arbeitsweise eines Kellerautomaten



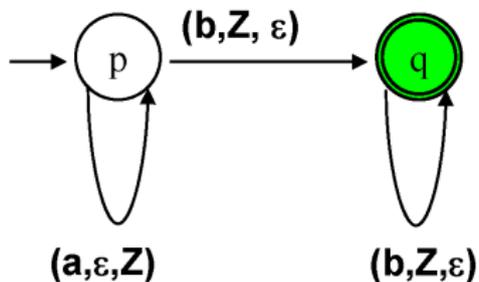
(noch) zu lesendes Wort:

b b
-----

aktueller Stack:

Z
Z

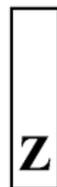
## Arbeitsweise eines Kellerautomaten



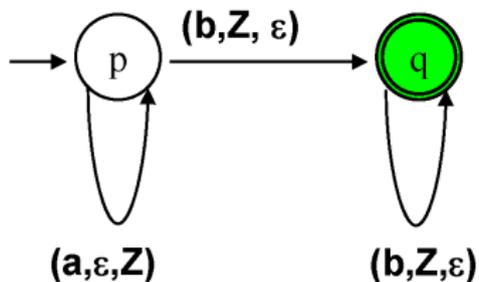
(noch) zu lesendes Wort:



aktueller Stack:



## Arbeitsweise eines Kellerautomaten



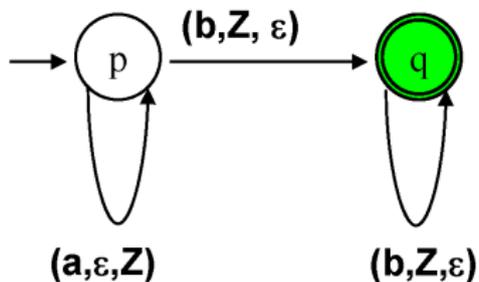
(noch) zu lesendes Wort:



aktueller Stack:



## Arbeitsweise eines Kellerautomaten



(noch) zu lesendes Wort:

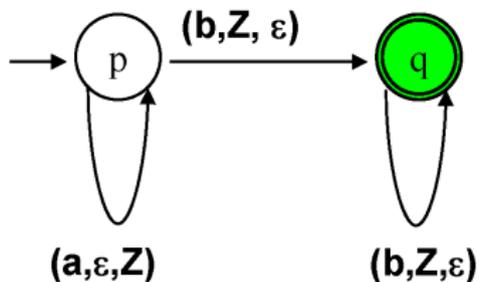


Das Wort ist abgearbeitet!

aktueller Stack:



## Arbeitsweise eines Kellerautomaten



(noch) zu lesendes Wort:



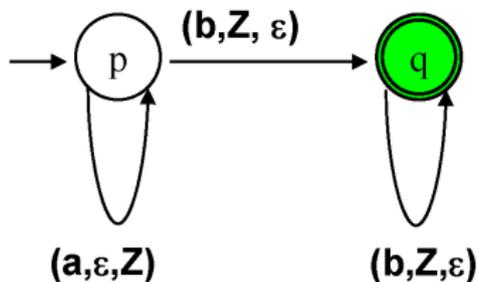
Das Wort ist abgearbeitet!

aktueller Stack:



Der Stack ist leer!

## Arbeitsweise eines Kellerautomaten



Der Automat befindet sich in einem Endzustand!

(noch) zu lesendes Wort:



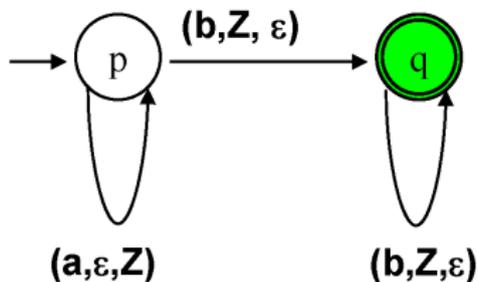
Das Wort ist abgearbeitet!

aktueller Stack:



Der Stack ist leer!

## Arbeitsweise eines Kellerautomaten



Der Automat befindet sich in einem Endzustand!

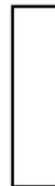
Darum akzeptiert der Kellerautomat das Wort!

(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 ( $q_0$ ) darin besteht, das Startsymbol  $S$  der Grammatik in den Keller zu legen.
- Während der eigentlichen Rechnung befindet sich der Automat permanent in dem anderen Zustand ( $q_1$ ), dem einzigen Endzustand; die Rechnung findet nur in dem Keller statt.
- Der konstruierte Automat vollzieht zwei verschiedene Arbeitsschritte:
  - Nicht-Leseschritt mit Bezug auf Grammatikregel  $B \rightarrow \beta$  (Expansion):  
Ersetze die Kellerspitze  $B$  mit  $\beta$
  - Leseschritte (Scan):  
Lies ein  $a \in T$  der Eingabekette und entferne  $a$  von der Kellerspitze.

# Beispiel: Von kontextfreien Grammatiken zu Kellerautomaten

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

