Prof. Dr. J. Giesl

M. Brockschmidt, F. Emmes, C. Fuhs, C. Otto, T. Ströder

Hinweise:

- Die **Hausaufgaben** sollen in Gruppen von je 2 Studierenden aus dem gleichen Tutorium bearbeitet werden.
- Die Lösungen der Hausaufgaben müssen bis Mi., 30.06.2010 im Tutorium abgegeben werden. Alternativ ist es bis 17 Uhr möglich, diese in den Kasten im Flur des LuFG I2 einzuwerfen (Ahornstr. 55, E1, 2. Etage).
- Namen und Matrikelnummern der Studierenden sowie **die Nummer der Übungsgruppe** sind auf jedes Blatt der Abgabe zu schreiben. **Heften bzw. tackern Sie die Blätter**!
- Die **Tutoraufgaben** werden in den jeweiligen Tutorien gemeinsam besprochen und bearbeitet.

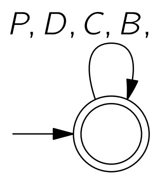
Tutoraufgabe 1 (ϵ-Produktionen):

Überführen Sie die folgende Grammatik mit dem in der Vorlesung vorgestellten Verfahren in eine äquivalente Grammatik ohne ϵ-Produktionen.

$$
\begin{aligned}
S &\to HB \\
H &\to xTLy \\
T &\to t \mid \epsilon \\
L &\to LL \mid \ell \mid \epsilon \\
B &\to iCj \mid \epsilon \\
C &\to CC \mid D \mid P \\
D &\to DD \mid P \mid \epsilon \\
P &\to p \mid \epsilon
\end{aligned}
$$

Lösung:

Der Automat für $pre^*(\{\epsilon\})$:



Es sind also P, D, C, B, L und T nullierbar. Wir fügen die Produktionen $B \to ij$, $L \to L$, $C \to C$, $D \to D$, $H \to xTy$, $H \to xLy$, $H \to xy$ und $S \to H$ hinzu und löschen danach $T \to \epsilon$, $L \to \epsilon$, $B \to \epsilon$, $D \to \epsilon$ und $P \to \epsilon$. Es ergibt sich eine Grammatik mit den folgenden Produktionsregeln:

$$
\begin{aligned}
S &\rightarrow HB \mid H \\
H &\rightarrow xTLy \mid xTy \mid xLy \mid xy \\
T &\rightarrow t \\
L &\rightarrow LL \mid \ell \mid L \\
B &\rightarrow iCj \mid ij \\
C &\rightarrow CC \mid D \mid P \mid C \\
D &\rightarrow DD \mid P \mid D \\
P &\rightarrow p
\end{aligned}
$$

Hausaufgabe 2 (ϵ-Produktionen): (2 Punkte)

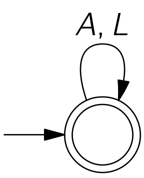
Überführen Sie die folgende Grammatik mit dem in der Vorlesung vorgestellten Verfahren in eine äquivalente Grammatik ohne ϵ-Produktionen.

$$
\begin{aligned}
S &\rightarrow (S+S) \mid (L-S) \mid N \\
N &\rightarrow DA \\
D &\rightarrow 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \\
A &\rightarrow AA \mid 0 \mid D \mid \epsilon \\
L &\rightarrow S \mid \epsilon
\end{aligned}
$$

Hinweis: "(" und ")" sind Terminalsymbole.

Lösung:

Der Automat für $pre^*(\{\epsilon\})$:



Es sind also A und L nullierbar. Wir fügen die Produktionen $S \rightarrow (-S)$, $A \rightarrow A$ und $N \rightarrow D$ hinzu und löschen $A \rightarrow \epsilon$ und $L \rightarrow \epsilon$. Es ergibt sich eine Grammatik mit den folgenden Produktionsregeln:

$$
\begin{aligned}
S &\rightarrow (S+S) \mid (L-S) \mid (-S) \mid N \\
N &\rightarrow DA \mid D \\
D &\rightarrow 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \\
A &\rightarrow AA \mid 0 \mid D \mid A \\
L &\rightarrow S
\end{aligned}
$$

Tutoraufgabe 3 (Chomsky-Normalform):

Überführen Sie die folgende Grammatik mit dem in der Vorlesung vorgestellten Verfahren in eine äquivalente Grammatik in Chomsky-Normalform.

$$\begin{array}{rcl} S & \rightarrow & aB \mid bA \mid ABc \mid B \\ A & \rightarrow & SSa \\ B & \rightarrow & cS \mid bB \mid b \end{array}$$

Lösung:

$$\begin{array}{rcl} S & \rightarrow & R_aB \mid R_aR_b \mid R_bA \mid AC \mid R_cS \mid R_cB \mid R_cR_b \mid R_bB \mid R_bR_b \mid b \\ A & \rightarrow & SD \mid BD \mid R_bD \\ B & \rightarrow & R_cS \mid R_cB \mid R_cR_b \mid R_bB \mid R_bR_b \\ C & \rightarrow & BR_c \mid R_bR_c \\ D & \rightarrow & SR_a \mid BR_a \mid R_bR_a \\ R_a & \rightarrow & a \\ R_b & \rightarrow & b \\ R_c & \rightarrow & c \end{array}$$

Hausaufgabe 4 (Chomsky-Normalform): (4 Punkte)

Überführen Sie die folgende Grammatik mit dem in der Vorlesung vorgestellten Verfahren in eine äquivalente Grammatik in Chomsky-Normalform.

$$\begin{array}{rcl} S & \rightarrow & aSbS \mid C \\ C & \rightarrow & cC \mid c \end{array}$$

Lösung:

$$
\begin{array}{rcl}
S & \to & R_a A \mid R_c C \mid R_c R_c \mid c \\
A & \to & SB \mid CB \mid R_c B \\
B & \to & R_b S \mid R_b C \mid R_b R_c \\
C & \to & R_c C \mid R_c R_c \\
R_a & \to & a \\
R_b & \to & b \\
R_c & \to & c
\end{array}
$$

Tutoraufgabe 5 (CYK-Algorithmus):

Gegeben sei die folgende Grammatik G.

$$
\begin{array}{rcl}
S & \to & SS \mid R_a A \mid R_b B \mid R_c C \\
A & \to & R_b R_c \mid R_c R_b \\
B & \to & R_a R_c \mid R_c R_a \\
C & \to & R_a R_b \mid R_b R_a \\
R_a & \to & a \\
R_b & \to & b \\
R_c & \to & c
\end{array}
$$

Testen Sie mit dem CYK-Algorithmus, ob die folgenden Wörter von G erzeugt werden.

a) $w_1 = abccab$

b) $w_2 = abcba$

Lösung:

a)

$w =$	a	b	c	c	a	b
1	R_a	R_b	R_c	R_c	R_a	R_b
2	C	A		B	C	
3	S			S		
4						
5						
6	S					

$\Rightarrow w_1 \in L(G)$

b)

$w =$	a	b	c	b	a
1	R_a	R_b	R_c	R_b	R_a
2	C	A	A	C	
3	S		S		
4					
5					

$\Rightarrow w_2 \notin L(G)$

Hausaufgabe 6 (CYK-Algorithmus): (3 + 3 = 6 Punkte)

Gegeben sei die folgende Grammatik G.

$$
\begin{aligned}
S &\rightarrow R_aA \mid R_eB \mid R_gC \mid R_lD \mid R_rE \mid R_aR_a \mid R_eR_e \mid R_gR_g \mid R_lR_l \mid R_rR_r \\
A &\rightarrow SR_a \\
B &\rightarrow SR_e \\
C &\rightarrow SR_g \\
D &\rightarrow SR_l \\
E &\rightarrow SR_r \\
R_a &\rightarrow a \\
R_e &\rightarrow e \\
R_g &\rightarrow g \\
R_l &\rightarrow l \\
R_r &\rightarrow r
\end{aligned}
$$

Testen Sie mit dem CYK-Algorithmus, ob die folgenden Wörter von G erzeugt werden.

a) $w_1 = regallager$

b) $w_2 = allee$

Lösung:

a)

$w =$	r	e	g	a	l	l	a	g	e	r
1	R_r	R_e	R_g	R_a	R_l	R_l	R_a	R_g	R_e	R_r
2					S					
3					A					
4				S						
5				C						
6			S							
7			B							
8		S								
9		E								
10	S									

$\Rightarrow w_1 \in L(G)$

b)

$w =$	a	l	l	e	e
1	R_a	R_l	R_l	R_e	R_e
2		S		S	
3		B			
4					
5					

$\Rightarrow w_2 \notin L(G)$

Tutoraufgabe 7 (Greibach-Normalform):

Verwenden Sie den Algorithmus aus der Vorlesung, um für die folgende Grammatik G eine Grammatik G' in Greibach-Normalform mit $L(G') = L(G)$ zu bestimmen. Verwenden Sie dabei die folgende Ordnung der Nichtterminale: $S < A < B < C$.

$$
\begin{aligned}
S &\to AS \mid SB \\
A &\to a \mid CB \\
B &\to AC \mid b \\
C &\to AB
\end{aligned}
$$

Lösung:

Die Regel $B \to AC$ ist die echt verkleinernde Regel mit dem kleinsten linken Nichtterminal und wird ersetzt.

$$
\begin{aligned}
S &\to AS \mid SB \\
A &\to a \mid CB \\
B &\to aC \mid CBC \mid b \\
C &\to AB
\end{aligned}
$$

Die Regel $C \to AB$ ist die echt verkleinernde Regel mit dem kleinsten linken Nichtterminal und wird ersetzt.

$$
\begin{aligned}
S &\to AS \mid SB \\
A &\to a \mid CB \\
B &\to aC \mid CBC \mid b \\
C &\to aB \mid CBB
\end{aligned}
$$

Nun gibt es keine echt verkleinernden Regeln mehr. Die Linksrekursion in den S-Regeln wird beseitigt. Dabei wird das Nichtterminal Z als neues kleinstes Element eingeführt.

$$
\begin{aligned}
Z &\to B \mid BZ \\
S &\to AS \mid ASZ \\
A &\to a \mid CB \\
B &\to aC \mid CBC \mid b \\
C &\to aB \mid CBB
\end{aligned}
$$

Die Linksrekursion in den C-Regeln wird beseitigt. Dabei wird das Nichtterminal Y als neues kleinstes Element eingeführt.

$$
\begin{aligned}
Y &\to BB \mid BBY \\
Z &\to B \mid BZ \\
S &\to AS \mid ASZ \\
A &\to a \mid CB \\
B &\to aC \mid CBC \mid b \\
C &\to aB \mid aBY
\end{aligned}
$$

Nun gibt es keine verkleinernde Regel mehr. Es bleibt, die auf manchen rechten Seiten führenden Nichtterminale zu eliminieren. Das größte Nichtterminal (C) ist nach Konstruktion schon in Greibach Form. Wir beginnen also mit B.

$$\begin{aligned}
Y &\to BB \mid BBY \\
Z &\to B \mid BZ \\
S &\to AS \mid ASZ \\
A &\to a \mid CB \\
B &\to aC \mid aBBC \mid aBYBC \mid b \\
C &\to aB \mid aBY
\end{aligned}$$

Nun folgt die Regel $A \to CB$.

$$\begin{aligned}
Y &\to BB \mid BBY \\
Z &\to B \mid BZ \\
S &\to AS \mid ASZ \\
A &\to a \mid aBB \mid aBYB \\
B &\to aC \mid aBBC \mid aBYBC \mid b \\
C &\to aB \mid aBY
\end{aligned}$$

Und nun die S-Regeln.

$$\begin{aligned}
Y &\to BB \mid BBY \\
Z &\to B \mid BZ \\
S &\to aS \mid aBBS \mid aBYBS \mid aSZ \mid aBBSZ \mid aBYBSZ \\
A &\to a \mid aBB \mid aBYB \\
B &\to aC \mid aBBC \mid aBYBC \mid b \\
C &\to aB \mid aBY
\end{aligned}$$

Nach dem Ersetzen in den Z- und Y-Regeln befindet sich die Grammatik in Greibach-Normalform:

$$\begin{aligned}
Y &\to aCB \mid aBBCB \mid aBYBCB \mid bB \mid aCBY \mid aBBCBY \mid aBYBCBY \mid bBY \\
Z &\to aC \mid aBBC \mid aBYBC \mid bB \mid aCZ \mid aBBCZ \mid aBYBCZ \mid bZ \\
S &\to aS \mid aBBS \mid aBYBS \mid aSZ \mid aBBSZ \mid aBYBSZ \\
A &\to a \mid aBB \mid aBYB \\
B &\to aC \mid aBBC \mid aBYBC \mid b \\
C &\to aB \mid aBY
\end{aligned}$$

______________________________.

Hausaufgabe 8 (Greibach-Normalform): (5 Punkte)

Verwenden Sie den Algorithmus aus der Vorlesung, um für die folgende Grammatik G eine Grammatik G' in Greibach-Normalform mit $L(G') = L(G)$ zu bestimmen. Verwenden Sie dabei die folgende Ordnung der Nichtterminale: $S < A < B < C$.

$$\begin{aligned}
S &\to AB \mid a \\
A &\to SA \\
B &\to CA \mid b \\
C &\to BB \mid c
\end{aligned}$$

Lösung:

Die Regel $A \to SA$ ist die echt verkleinernde Regel mit dem kleinsten linken Nichtterminal und wird ersetzt.

$$\begin{aligned} S &\to AB \mid a \\ A &\to ABA \mid aA \\ B &\to CA \mid b \\ C &\to BB \mid c \end{aligned}$$

Die Regel $C \to BB$ ist die echt verkleinernde Regel mit dem kleinsten linken Nichtterminal und wird ersetzt.

$$\begin{aligned} S &\to AB \mid a \\ A &\to ABA \mid aA \\ B &\to CA \mid b \\ C &\to CAB \mid bB \mid c \end{aligned}$$

Nun gibt es keine echt verkleinernden Regeln mehr. Die Linksrekursion in den A-Regeln wird beseitigt. Dabei wird das Nichtterminal Z als neues kleinstes Element eingeführt.

$$\begin{aligned} Z &\to BA \mid BAZ \\ S &\to AB \mid a \\ A &\to aA \mid aAZ \\ B &\to CA \mid b \\ C &\to CAB \mid bB \mid c \end{aligned}$$

Die Linksrekursion in den C-Regeln wird beseitigt. Dabei wird das Nichtterminal Y als neues kleinstes Element eingeführt.

$$\begin{aligned} Y &\to AB \mid ABY \\ Z &\to BA \mid BAZ \\ S &\to AB \mid a \\ A &\to aA \mid aAZ \\ B &\to CA \mid b \\ C &\to bB \mid c \mid bBY \mid cY \end{aligned}$$

Nun gibt es keine verkleinernde Regel mehr. Es bleibt, die auf manchen rechten Seiten führenden Nichtterminale zu eliminieren. Das größte Nichtterminal (C) ist nach Konstruktion schon in Greibach Form. Wir beginnen also mit B.

$$\begin{aligned} Y &\to AB \mid ABY \\ Z &\to BA \mid BAZ \\ S &\to AB \mid a \\ A &\to aA \mid aAZ \\ B &\to bBA \mid cA \mid bBYA \mid cYA \mid b \\ C &\to bB \mid c \mid bBY \mid cY \end{aligned}$$

Nun folgt S, da die A-Regeln ebenfalls schon die gewünschte Form haben.

$$\begin{aligned}
Y &\to AB \mid ABY \\
Z &\to BA \mid BAZ \\
S &\to aAB \mid aAZB \mid a \\
A &\to aA \mid aAZ \\
B &\to bBA \mid cA \mid bBYA \mid cYA \mid b \\
C &\to bB \mid c \mid bBY \mid cY
\end{aligned}$$

Nach dem Ersetzen in den Z- und Y-Regeln befindet sich die Grammatik in Greibach-Normalform:

$$\begin{aligned}
Y &\to aAB \mid aAZB \mid aABY \mid aAZBY \\
Z &\to bBAB \mid cAB \mid bBYAB \mid cYAB \mid bA \mid bBAAZ \mid cAAZ \mid bBYAAZ \mid cYAAZ \mid bAZ \\
S &\to aAB \mid aAZB \mid a \\
A &\to aA \mid aAZ \\
B &\to bBA \mid cA \mid bBYA \mid cYA \mid b \\
C &\to bB \mid c \mid bBY \mid cY
\end{aligned}$$

Tutoraufgabe 9 (Pumping-Lemma):

Prüfen Sie, ob folgende Sprachen über dem Alphabet $\Sigma = \{a, b, c\}$ kontextfrei sind. Falls die Sprache nicht kontextfrei ist, beweisen Sie dies mit Hilfe des Pumping-Lemmas für kontextfreie Sprachen. Falls die Sprache kontextfrei ist, geben Sie eine CFG an, welche die Sprache erzeugt.

a) $L_1 = \{(ab)^n(ca)^n(bc)^n \mid n \geq 0\}$

b) $L_2 = \{a^n b^{2n} c \mid n \geq 0\}$

Lösung:

a) Sei $n \in \mathbb{N}$. Wir wählen das Wort $z = (ab)^n(ca)^n(bc)^n \in L_1$. Sei $uvwxy = z$ eine Zerlegung mit $|vwx| \leq n$ und $|vx| > 0$. Wir wählen nun $i = 0$ und betrachten das Wort $z' = uwy$. Wegen $|vwx| \leq n$ gibt es nur zwei Fälle. Falls $(bc)^n$ ein Teilwort von y ist, muss z' zu wenige ab oder ca Teilworte enthalten und es gilt $z' \notin L_1$. Falls $(ab)^n$ ein Teilwort von u ist, muss z' zu wenige ca oder bc Teilworte enthalten und es gilt wiederum $z' \notin L_1$. Damit ist L_1 nicht kontextfrei.

b) L_2 ist kontextfrei mit $L_2 = L(G)$ für G wie folgt:

$$\begin{aligned}
S &\to Ac \\
A &\to aAbb \mid \epsilon
\end{aligned}$$

Hausaufgabe 10 (Pumping-Lemma): (3 + 3 + 3 = 9 Punkte)

Prüfen Sie, ob folgende Sprachen über dem Alphabet $\Sigma = \{a, b, c\}$ kontextfrei sind. Falls die Sprache nicht kontextfrei ist, beweisen Sie dies mit Hilfe des Pumping-Lemmas für kontextfreie Sprachen. Falls die Sprache kontextfrei ist, geben Sie eine CFG an, welche die Sprache erzeugt. Wir erinnern an die Funktionen $\sharp_z$ für ein Zeichen $z \in \Sigma$, welche die Anzahl der Zeichen z in einem Wort zählen.

a) $L_3 = \{a^n b^{2n} c^{3n} \mid n \geq 0\}$

b) $L_4 = \{w \in \Sigma^* \mid \sharp_a(w) = \sharp_b(w) = \sharp_c(w)\}$

c) $L_5 = \{a^n b^{n^2} \mid n \geq 0\}$

Lösung:

a) Sei $n \in \mathbb{N}$. Wir wählen das Wort $z = a^n b^{2n} c^{3n} \in L_3$. Sei $uvwxy = z$ eine Zerlegung mit $|vwx| \leq n$ und $|vx| > 0$.

Wir wählen nun $i = 0$ und betrachten damit das Wort $z' = uwy$. Wegen $|vwx| \leq n$ gibt es nur zwei Fälle.

Fall $vwx = a^k b^\ell$ für $k, \ell \in \mathbb{N}_0$: Da $|vx| > 0$ gilt, muss mindestens eins von beiden nicht-leer sein. Durch das Löschen von v und x aus z erhalten wir also ein Wort, das weiterhin mit c^{3n} endet, die Zahl der a oder b ist aber strikt kleiner geworden und daher gilt $z' \notin L_3$.

Fall $vwx = b^k c^\ell$ für $k, \ell \in \mathbb{N}_0$: Analog zum vorherigen Fall enthält z' mindestens ein b oder c zu wenig und wieder gilt $z' \notin L_3$.

Die Fälle $vwx = a^k$, $vwx = b^k$ und $vwx = c^k$ sind in den beiden betrachteten Fällen enthalten, da dort k bzw. ℓ jeweils als 0 gewählt werden können.

Die Bedingungen des Pumping Lemmas gelten also nicht für die Sprache L_3 und das Wort z. Daher ist L_3 keine kontextfreie Sprache.

b) Sei $n \in \mathbb{N}$. Wir wählen das Wort $z = a^n b^n c^n \in L_4$. Sei $uvwxy = z$ eine Zerlegung mit $|vwx| \leq n$ und $|vx| > 0$.

Wir wählen nun $i = 0$ und betrachten das Wort $z' = uwy$. Wegen $|vwx| \leq n$ gibt es nur zwei Fälle.

Fall $vwx = a^k b^\ell$ für $k, \ell \in \mathbb{N}_0$: Da $|vx| > 0$ gilt, muss mindestens eins von beiden nicht-leer sein. Durch das Löschen von v und x aus z erhalten wir also ein Wort z', das weiterhin mit c^n endet, die Zahl der a oder b ist aber strikt kleiner geworden und daher gilt $z' \notin L_4$.

Fall $vwx = b^k c^\ell$ für $k, \ell \in \mathbb{N}_0$: Analog zum vorherigen Fall enthält z' mindestens ein b oder c zu wenig und wieder gilt $z' \notin L_4$.

Die Fälle $vwx = a^k$, $vwx = b^k$ und $vwx = c^k$ sind in den beiden betrachteten Fällen enthalten, da dort k bzw. ℓ jeweils als 0 gewählt werden können.

Die Bedingungen des Pumping Lemmas gelten also nicht für die Sprache L_4 und das Wort z. Daher ist L_4 keine kontextfreie Sprache.

c) Sei $n \in \mathbb{N}$. Wir wählen das Wort $z = a^n b^{n^2} \in L_5$. Sei $uvwxy = z$ eine Zerlegung mit $|vwx| \leq n$ und $|vx| > 0$.

Wir wählen nun $i = 2$ und betrachten das Wort $z' = uv^2wx^2y$. Falls $vwx \in L(a^*)$ oder $vwx \in L(b^*)$, gilt direkt $z' \notin L_5$, da nur die Anzahl von a's oder b's verändert wird und die Bedingung $\sharp_a(z') = \sharp_b(z')^2$ verletzt wird.

Sei also $vwx = a^k b^\ell$. Falls $v = a^k b^t$ oder $x = a^t b^\ell$ ist, gilt $z' = uvvwxxy = ua^k b^t a^k b^t wxxy$ bzw. $z' = uvvwa^t b^\ell a^t b^\ell y$, d.h. z' enthält das Teilwort ba und damit ist $z' \notin L_5$.

Im letzten verbleibenden Fall gilt $v = a^p$ und $x = b^q$ für $p, q \in \mathbb{N}_0$. Dann ist $w = a^{k-p}b^{\ell-q}$ und $z' = uvvwxxy = a^{n-k}a^pa^pa^{k-p}b^{\ell-q}b^qb^qb^{n^2-q} = a^{n+p}b^{n^2+q}$. Es gilt $(n+p)^2 = n^2+2np+p^2$. Damit $z' \in L_5$ gilt, müsste also $n^2+2np+p^2 = n^2+q \Leftrightarrow 2np+p^2 = q$ gelten. Wegen $n \geq |vwx| \geq q$ ist dies ein Widerspruch und es gilt $z' \notin L_5$. Damit ist L_5 nicht kontextfrei.

Hausaufgabe 11 (Pumping-Lemma): (2 Punkte)

Zeigen Sie, dass das Pumping-Lemma für kontextfreie Sprachen für jede endliche Sprache erfüllt ist. Beweisen Sie dazu die folgende Aussage:

Sei L eine endliche Sprache. Dann existiert ein $n \in \mathbb{N}$, sodass für jedes $z \in L$ mit $|z| \geq n$ gilt, dass Wörter u, v, w, x, y existieren mit $z = uvwxy$, $|vwx| \leq n$ und $|vx| > 0$, sodass für jedes $i \in \mathbb{N}$ gilt, dass $uv^iwx^iy \in L$.

Sie dürfen in diesem Beweis nicht verwenden, dass *jede* kontextfreie Sprache das Pumping-Lemma für kontextfreie Sprachen erfüllt.

Lösung:

Sei L eine endliche Sprache. Sei $n \in \mathbb{N}$ größer als die Länge aller Wörter in L. Dann gibt es kein Wort $z \in L$ mit $|z| \geq n$ und die Aussage gilt trivialerweise.

Alternativer Beweis mit Hilfe des Pumping-Lemmas für reguläre Sprachen:
Sei L eine endliche Sprache. Da L regulär ist, gilt nach dem Pumping-Lemma für reguläre Sprachen, dass ein $n \in \mathbb{N}$ existiert, sodass für jedes $z \in L$ mit $|z| \geq n$ gilt, dass Wörter r, s, t existieren mit $z = rst$, $|rs| \leq n$ und $|s| > 0$, sodass für jedes $i \in \mathbb{N}$ gilt, dass $rs^it \in L$. Wähle $u = v = \epsilon$, $w = r$, $x = s$ und $y = t$. Dann gilt $|vwx| = |rs| \leq n$, $|vx| = |s| > 0$ und $uv^iwx^iy = rs^it \in L$ für alle $i \in \mathbb{N}$. Damit ist die Aussage bewiesen.