FGI-1 – Formale Grundlagen der Informatik I

Logik, Automaten und Formale Sprachen

Musterlösung 4 : Kontextfreie Grammatiken und Kellerautomaten

Präsenzaufgabe 4.1: Sei $L = \{a^{k+l}b^kc^l \mid k, l \geq 1\}$.

1. Konstruieren Sie eine kontextfreie Grammatik $G = (\Sigma, N, P, S)$, so dass $L(G) = L$.

   Lösung: Eine lineare Grammatik ist:

$$\begin{aligned} S &\to aCc \\ C &\to aCc \mid B \\ B &\to aBb \mid ab \end{aligned}$$

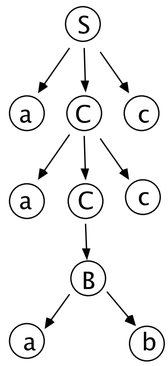
2. Ist Ihre Grammatik linear? Wenn nein, konstruieren Sie eine.
3. Konstruieren Sie eine Ableitung für das Wort $w = aaabcc$. Ist Ihre Ableitung eine Linksableitung? Eine Rechtsableitung?

   Lösung: Eine (Links-)Ableitung ist:

$$S \Longrightarrow aCc \Longrightarrow aaCcc \Longrightarrow aaBcc \Longrightarrow aaabcc$$

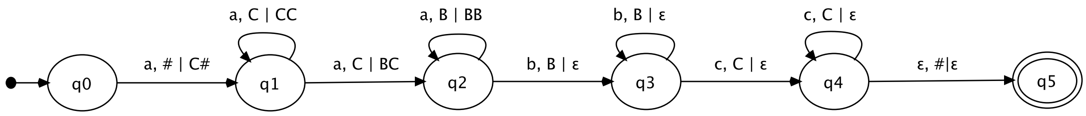
4. Konstruieren Sie einen Ableitungsbaum für $w = aaabcc$.

   Lösung:



5. Konstruieren Sie einen PDA A mit $L(A) = L$.

   Lösung: Der folgende PDA für $L = \{a^{k+l}b^kc^l \mid k, l \geq 1\}$ merkt sich zunächst die Anzahl $l > 0$ im Zustand q_1 und danach die Anzahl $k > 0$ im Zustand q_2. Hierbei ist wichtig, dass in q_1 im richtigen Moment geraten wird, von k auf l zu wechseln, d.h. von q_1 nicht mehr wieder nach q_1, sondern nach q_2 zu wechseln. Im Zustand q_3 wird dann der Keller wieder abgebaut und zwar um genau die k-fach vorhandenen B. Hierbei wird jeweils ein b gelesen. Im Zustand q_4 wird dann der Keller wieder abgebaut und zwar um genau die l-fach vorhandenen C. Hierbei wird jeweils ein c gelesen. Genau dann, wenn der Keller komplett geleert wurde und das Kellerbodenzeichen $\#$ sichtbar wird, geht der PDA in den Endzustand q_5 über, wobei auch noch der Keller geleert wird. Wir haben somit eine Möglichkeit, jedes Wort aus L zu akzeptieren: $L \subseteq L(A)$.



Umgekehrt sehen wir sofort, dass jedes akzeptierbare Wort aus $\{a\}\{a\}^+\{b\}^+\{c\}^+$ sein muss, da wir entlang eines Pfades im Zustanddiagramm, der uns von q_0 zu q_5 führt, nur solche Eingaben lesen können. Wird ein Wort $w = a^i b^j c^k$ akzeptiert gilt also $i > 1, j > 0, k > 0$. Wir lesen weiterhin ab, dass wir für jedes gelesene a ein Symbol auf den Keller hinzufügen. Da wir den Übergang von q_4 nach q_5 nur machen können, wenn wir das Kellerbodenzeichen lesen, muss also $|w|_a = |w|_b + |w|_c$ gelten, womit folgt, dass $i = j + k$ gilt, d.h. $w \in L$. Also $L(A) \subseteq L$.

6. Geben Sie eine Erfolgsrechung für $w = aaabcc$ an.

Lösung:

$$
\begin{aligned}
&(q_0, aaabcc, \perp)\\
&\vdash (q_1, aabcc, C\perp)\\
&\vdash (q_1, abcc, CC\perp)\\
&\vdash (q_2, bcc, BCC\perp)\\
&\vdash (q_3, cc, CC\perp)\\
&\vdash (q_4, c, C\perp)\\
&\vdash (q_4, \epsilon, \perp)\\
&\vdash (q_5, \epsilon, \epsilon)
\end{aligned}
$$

Präsenzaufgabe 4.2:

1. Gegeben sei die Grammatik $G = (\Sigma, N, P, S)$ mit den Produktionen:

$$S \to aSb \mid ab$$

Beweisen Sie, (a) $L(G) \subseteq M := \{a^n b^n \mid n \geq 1\}$ und (b) $M \subseteq L(G)$.

Lösung:

(a) $L(G) \subseteq M := \{a^n b^n \mid n \geq 1\}$

Wir zeigen per Induktion über die Länge n der Ableitung, dass für jedes n gilt: Wenn $S \stackrel{n}{\Longrightarrow} u$ gilt, dann gilt entweder $u = a^n S b^n$ oder $u = a^n b^n$.

- Ind.Anfang $n = 1$. Die einzig möglichen Ableitungen der Länge $n = 1$ sind $S \Longrightarrow aSb$ und $S \Longrightarrow ab$. Also hat u die Form $u = a^1 S b^1 \in M$ bzw. $u = a^1 b^1 \in M$ und die Behauptung gilt.
- Ind.Annahme: Die Behauptung gelte für alle Ableitungen Länge kleiner oder gleich n.
- Ind.Schritt: Eine Ableitung der Länge $n + 1$ hat die Form $S \stackrel{n}{\Longrightarrow} u \Longrightarrow v$. Nach IA gilt entweder $u = a^n S b^n$ oder $u = a^n b^n$. Da u noch eine Ableitung erlaubt, muss $u = a^n S b^n$ sein.
  Beide Produktionen sind anwendbar: $u = a^n S b^n \Longrightarrow a^n(aSb)b^n = a^{n+1} S b^{n+1}$ bzw. $u = a^n S b^n \Longrightarrow a^n(ab)b^n = a^{n+1} b^{n+1}$. Dies ist die Behauptung für $n + 1$.

Somit gilt die Ind.Behauptung für alle n.

Also gilt: Wenn ein Terminalwort $S \stackrel{*}{\Longrightarrow} w$ ableitbar ist, dann existiert ein n, so dass $S \stackrel{n}{\Longrightarrow} w$ gilt und dann ist mit obiger Aussage $w = a^n b^n$ und damit in M.

(b) $M \subseteq L(G)$

Sei $w = a^n b^n \in M$ für ein beliebiges, aber festes $n \geq 1$. Dann kann w folgendermaßen abgeleitet werden:

$$\underbrace{S \Longrightarrow aSb \Longrightarrow \cdots \Longrightarrow a^{n-1} S b^{n-1}}_{(n-1)\text{-mal}} \Longrightarrow a^n b^n$$

Dieses Argument ist unmittelbar einleuchtend, kann aber noch durch Induktion formalisiert werden.

2. Reduzieren Sie mit dem Verfahren der Vorlesung die folgende Grammatik $G = (\Sigma, N, P, S)$.

$$\begin{aligned}
&S \to Ca\\
&A \to bA \mid B\\
&B \to aBb\\
&C \to Cac \mid CC \mid c\\
&D \to A \mid BA \mid a
\end{aligned}$$

Lösung:

(a) Produktive Symbole:

$$\begin{aligned}
M_0 &= \Sigma\\
M_1 &= M_0 \cup \{C, D\}\\
M_2 &= M_1 \cup \{S\}\\
M_3 &= M_2
\end{aligned}$$

Grammatik $G' = (\Sigma, N', P', S)$ mit $N' = \{S, C, D\}$ und:

$$
\begin{aligned}
&S \to Ca \\
&C \to Cac \mid CC \mid c \\
&D \to a
\end{aligned}
$$

(b) Erreichbare Symbole:

$$
\begin{aligned}
M_0 &= \{S\} \\
M_1 &= M_0 \cup \{C\} \\
M_2 &= M_1
\end{aligned}
$$

Grammatik $G'' = (\Sigma, N'', P', S)$ mit $N' = \{S, C\}$ und:

$$
\begin{aligned}
&S \to Ca \\
&C \to Cac \mid CC \mid c
\end{aligned}
$$

Übungsaufgabe 4.3: KFG, PDA.

von
6

1. Ein Palindrom ist ein Wort w, das vorwärts und rückwärts gelesen gleich ist: $w = w^{rev}$. Konstruieren Sie einen Kellerautomaten A mit

$$L(A) = L := \{w \in \{a, b\}^* \mid w \text{ ist ein Palindrom } \}$$

Argumentieren Sie schlüssig, warum Ihr Kellerautomat A alle Worte aus L akzeptiert und keine weiteren.

Lösung: Jedes Palindrom hat genau eine der beiden folgenden Formen: Entweder $|w|$ ist gerade, dann hat w die Form $w = uv$, wobei v das Wort u gespiegelt ist: $w = uu^{rev}$. Oder $|w|$ ist ungerade, dann hat w die Form $w = uau^{rev}$ bzw. $w = ubu^{rev}$.

Sei A ein Kellerautomat mit den 3 Zuständen q_0, q_1 und q_2. Sei q_0 der Start- und q_2 der Endzustand. Das Zustandsdiagramm sei definiert durch:

$q_0 \xrightarrow{a,\perp|A\perp} q_0, q_0 \xrightarrow{a,A|AA} q_0, q_0 \xrightarrow{a,B|AB} q_0,$ (Stack aufbauen: push(A).)
$q_0 \xrightarrow{b,\perp|B\perp} q_0, q_0 \xrightarrow{b,A|BA} q_0, q_0 \xrightarrow{b,B|BB} q_0,$ (Stack aufbauen: push(B).)
$q_0 \xrightarrow{\epsilon,\perp|\perp} q_1, q_0 \xrightarrow{\epsilon,A|A} q_1, q_0 \xrightarrow{\epsilon,B|B} q_1,$ (Gerades Wort: Zustandswechsel)
$q_0 \xrightarrow{a,A|A} q_1, q_0 \xrightarrow{a,B|B} q_1, q_0 \xrightarrow{b,A|A} q_1, q_0 \xrightarrow{b,B|B} q_1,$ (Ungerades Wort: Mittelzeichen lesen.)
$q_1 \xrightarrow{a,A|\epsilon} q_1, q_1 \xrightarrow{b,B|\epsilon} q_1,$ (Stack abbbauen.)
$q_1 \xrightarrow{\epsilon,\perp|\epsilon} q_2$ (Keller lehren.)

- Jedes Palindrom aus L ist, kann A w wie folgt akzeptieren: Das Wort u wird im Zustand q_0 eingelesen, wobei eine Kopie auf dem Keller in Großbuchstaben gespeichert wird.

  Hat w eine ungerade Länge, dann wird mit $q_0 \xrightarrow{a,A|\epsilon} q_1$ bzw. mit $q_0 \xrightarrow{b,B|\epsilon} q_1$ in den Zustand q_1 gewechselt. Hat w eine gerade Länge, dann wird mit $q_0 \xrightarrow{\epsilon,\perp|\perp} q_1$, $q_0 \xrightarrow{\epsilon,A|A} q_1$ oder $q_0 \xrightarrow{\epsilon,B|B} q_1$ in den Zustand q_1 gewechselt (jenachdem mit welchem Zeichen u geendet hat).

  In q_1 wird dann das Wort u^{rev} gelesen, was problemlos möglich ist, da auf dem Keller auch die Kellersymbole in umgekehrter Reihenfolge stehen. Ist das Wort dann komplett gelesen, dann wird mit $q_1 \xrightarrow{\epsilon,\perp|\epsilon} q_2$ in den Endzustand gewechselt.

- Betrachten wir nun ein akzeptiertes Wort w. Das Wort muss A nach q_2 geführt haben, da dies der einzige Endzustand ist. Alle Rechnungen bleiben eine endliche Anzahl in q_1 (und lesen das Wort $x \in \Sigma^*$), wechseln dann nach q_2 (und lesen das Wort $y \in \{\epsilon\} \cup \Sigma$), bleiben dort eine endliche Anzahl von Schleifen (und lesen das Wort $z \in \Sigma^*$) und wechseln schließlich nach q_2 (wobei nichts von der Eingabe gelesen) wird. Die Eingabe hat also die Form $w = xyz$.

  Wir erreichen q_2 nur, wenn der der Keller komplett geleert wird. Der einzige Zustand, in dem etwas auf den Keller hinzugefügt wird, ist q_0 (nämlich immer ein Zeichen). Der einzige Zustand, in dem etwas vom den Keller entfernt wird (nämlich auch immer ein Zeichen), ist q_1. Also wissen wir, dass x und z die gleiche Länge besitzen.

  Da in q_1 jedes Eingabesymbol der als Großbuchstabe gemerkt wird und in q_2 nur dann ein Eingabesymbol gelesen wird, wenn der entsprechende Großbuchstabe oben auf dem Keller liegt, muss $z = x^{rev}$ gelten.

  Also hat ein akzeptiertes Wort w die Form xx^{rev}, xax^{rev} oder xbx^{rev}, was nach obiger Erläuterung ein Palindrom ist.

2. Konstruieren Sie eine kontextfreie Grammatik G mit $L(G) = L$, wobei gilt:

$$L = \{w \in \{a,b,c\}^* \mid \begin{array}{l}\text{auf einen a-Block folgen beliebig viele b's}\\ \text{und dann genausoviele c's wie zuvor a's}\}\end{array}$$

Beweisen Sie die Gleichheit $L(G) = L$, indem Sie zwei Mengeninklusionen zeigen.

Lösung: Jedes Wort $w \in L$ beginnt es mit einem Anfangsstück aus b's und c's, gefolgt von einem Wort $a^{n_1}b^{m_1}c^{n_1}$, dann wieder ein Teilwort aus b's und c's (das aber nicht mit c beginnen darf) usw.

Also gibt es ein $k \in \mathbb{N}$, Worte $u_0 \in \{b,c\}^*$ sowie $u_1, \dots, u_k \in \{b,c\}^* \setminus \{c\}^+ = \{\epsilon\} \cup \{b\}\{b,c\}^*$ und Längen $m_1 \dots, m_k \in \mathbb{N}$ sowie $n_1, \dots, n_k \in \mathbb{N} \setminus \{0\}$, so dass gilt:

$$w = u_0(a^{n_1}b^{m_1}c^{n_1})u_1 \cdots (a^{n_k}b^{m_k}c^{n_k})u_k$$

Wir definieren die Grammatik G folgendermaßen:

$$\begin{aligned}
&S \to XS'\\
&S' \to aAcYS' \mid \epsilon\\
&A \to aAc \mid B,\\
&B \to bB \mid \epsilon\\
&X \to bX \mid cX \mid \epsilon,\\
&Y \to bX \mid \epsilon,
\end{aligned}$$

(a) $L \subseteq L(G)$.
Jedes Wort $w \in L$ kann folgendermaßen abgeleitet werden:

$$\begin{aligned}
&S \overset{*}{\Longrightarrow} XS'\\
&\Longrightarrow X(aAc)YS'\\
&\overset{*}{\Longrightarrow} X(aAc)Y \cdots (aAc)YS'\\
&\Longrightarrow X(aAc)Y \cdots (aAc)Y\\
&\overset{*}{\Longrightarrow} u_0(aAc)Y \cdots (aAc)Y\\
&\overset{*}{\Longrightarrow} u_0(a^{n_1}Ac^{n_1}) \cdots (aAc)Y\\
&\Longrightarrow u_0(a^{n_1}Bc^{n_1})Y \cdots (aAc)Y\\
&\Longrightarrow u_0(a^{n_1}b^{m_1}c^{n_1})Y \cdots (aAc)Y\\
&\overset{*}{\Longrightarrow} u_0(a^{n_1}b^{m_1}c^{n_1})u_1 \cdots (aAc)Y\\
&\overset{*}{\Longrightarrow} u_0(a^{n_1}b^{m_1}c^{n_1})u_1 \cdots (aAc)Y\\
&\vdots\\
&\overset{*}{\Longrightarrow} u_0(a^{n_1}b^{m_1}c^{n_1})u_1 \cdots (a^{n_k}b^{m_k}c^{n_k})u_k
\end{aligned}$$

Dies ist unmittelbar einleuchtend, kann aber noch durch Induktion formalisiert werden.

(b) $L(G) \subseteq L$.
Betrachten wir obige Ableitung für w, dann stellen wir fest, dass dies die „einzige" Art und Weise ist, wie wir ein Wort ableiten können. Aus der Grammatik lesen wir ab:

- Zunächst stellen wir fest, dass jedes erzeugte B nur Worte der Form $b^m, m \in \mathbb{N}$ generieren kann.
- Dann stellen wir fest, dass ein A nur Worte der Form $a^nBc^n, n \in \mathbb{N}$ erzeugen kann.
- Insgesamt kann ein A nur Worte der Form $a^nb^mc^n, m,n \in \mathbb{N}$ erzeugen.
- Wir stellen fest, dass man mit dem Nonterminal X jedes Wort $u_0 \in \{b,c\}^*$ ableiten kann und mit Y jedes $u_1, \dots, u_k \in \{\epsilon\} \cup \{b\}\{b,c\}^*$.
- Aus S kann man (wenn man kein A weiter ableitet) nur Worte der Form $X(aAcY)^k, k \in \mathbb{N}$ erzeugen.

Setzen wir diese Argumente zusammen, dann stellen wir fest, dass *jede* (!) Ableitung zu einem Terminalwort die folgender Form hat:

$$S \overset{*}{\Longrightarrow} u_0 a^{n_1} b^{m_1} c^{n_1} u_1 \cdots a^{n_k} b^{m_k} c^{n_k} u_k$$

Alle diese Wörter sind in L.

Übungsaufgabe 4.4: Konstruieren Sie (mit dem Verfahren der Vorlesung) zu folgender Grammatik $G = (\Sigma, N, P, S)$ die Chomsky-Normalform!

von
6

$$\begin{aligned}
&S \to BCD \mid a\\
&A \to aBD \mid aDb \mid CSSD\\
&B \to bB \mid aC \mid bD\\
&C \to Cc \mid \epsilon\\
&D \to \epsilon \mid AC
\end{aligned}$$

Achten Sie auf die Nachvollziehbarkeit Ihrer Darstellung.

Lösung:

1. Elimantion der ϵ-Produktionen.

   Menge der Nonterminale, die ϵ erzeugen können:

$$\begin{aligned}
M_0 &= \{C, D\}\\
M_1 &= M_0
\end{aligned}$$

   Substitution:

$$\begin{aligned}
S &\to BCD \mid a\\
&\ \mathbf{BD} \mid \mathbf{BC} \mid \mathbf{B}\\
A &\to aBD \mid aDb \mid CSSD\\
&\ \mid \mathbf{aB} \mid \mathbf{ab} \quad \mid \mathbf{CSS} \mid \mathbf{SSD} \mid \mathbf{SS}\\
B &\to bB \mid aC \mid bD\\
&\ \mid \mathbf{a} \mid \mathbf{b}\\
C &\to Cc \mid \epsilon\\
&\ \mid \mathbf{c}\\
D &\to \epsilon \mid AC\\
&\ \mid \mathbf{A}
\end{aligned}$$

   Streichen der ϵ-Regeln ergibt

$$\begin{aligned}
S &\to BCD \mid a\\
&\ BD \mid BC \mid B\\
A &\to aBD \mid aDb \mid CSSD\\
&\ \mid aB \mid ab \mid CSS \mid SSD \mid SS\\
B &\to bB \mid aC \mid bD \mid a \mid b\\
C &\to Cc \mid c\\
D &\to AC \mid A
\end{aligned}$$

2. Reduktion: S, A, B und C sind unmittelbar produktiv; damit auch D. Von S sind alle B, C und D unmittelbar erreichbar; A ist über D erreichbar. Die Grammatik ist bereits reduziert.

3. Entfernung der Kettenregeln:

   Wir haben folgende Ketten $D \ll A$ und $S \ll B$.

$$\begin{aligned}
S &\to BCD \mid a\\
&\ BD \mid BC \qquad \mid \mathbf{bB} \mid \mathbf{aC} \mid \mathbf{bD} \mid \mathbf{a} \mid \mathbf{b}\\
A &\to aBD \mid aDb \mid CSSD\\
&\ \mid aB \mid ab \mid CSS \mid SSD \mid SS\\
B &\to bB \mid aC \mid bD \mid a \mid b\\
C &\to Cc \mid c\\
D &\to AC \mid \qquad \mathbf{aBD} \mid \mathbf{aDb} \mid \mathbf{CSSD} \mid \mathbf{aB} \mid \mathbf{ab} \mid \mathbf{CSS} \mid \mathbf{SSD} \mid \mathbf{SS}
\end{aligned}$$

4. Ersetzen langer Terminalregeln:

$$
\begin{array}{rcl}
S & \rightarrow & BCD \mid a \\
 & & BD \mid BC \mid \langle b\rangle B \mid \langle a\rangle C \mid \langle b\rangle D \mid a \mid b \\
A & \rightarrow & \langle a\rangle BD \mid \langle a\rangle D\langle b\rangle \mid CSSD \\
 & & \mid \langle a\rangle B \mid \langle a\rangle\langle b\rangle \mid CSS \mid SSD \mid SS \\
B & \rightarrow & \langle b\rangle B \mid \langle a\rangle C \mid \langle b\rangle D \mid a \mid b \\
C & \rightarrow & C\langle c\rangle \mid c \\
D & \rightarrow & AC \mid \langle a\rangle BD \mid \langle a\rangle D\langle b\rangle \mid CSSD \mid \langle a\rangle B \mid \langle a\rangle\langle b\rangle \mid CSS \mid SSD \mid SS \\
\langle a\rangle & \rightarrow & a \\
\langle b\rangle & \rightarrow & b \\
\langle c\rangle & \rightarrow & c
\end{array}
$$

5. Verkürzen langer Regeln:

$$
\begin{array}{rcl}
S & \rightarrow & \langle BC\rangle D \mid a \\
 & & BD \mid BC \mid \langle b\rangle B \mid \langle a\rangle C \mid \langle b\rangle D \mid a \mid b \\
A & \rightarrow & \langle\langle a\rangle B\rangle D \mid \langle\langle a\rangle D\rangle\langle b\rangle \mid \langle CSS\rangle D \\
 & & \mid \langle a\rangle B \mid \langle a\rangle\langle b\rangle \mid \langle CS\rangle S \mid \langle SS\rangle D \mid SS \\
B & \rightarrow & \langle b\rangle B \mid \langle a\rangle C \mid \langle b\rangle D \mid a \mid b \\
C & \rightarrow & C\langle c\rangle \mid c \\
D & \rightarrow & AC \mid \langle\langle a\rangle B\rangle D \mid \langle\langle a\rangle D\rangle\langle b\rangle \mid \langle CSS\rangle D \mid \langle a\rangle B \mid \langle a\rangle\langle b\rangle \mid \langle CS\rangle S \mid \langle SS\rangle D \mid SS \\
\langle a\rangle & \rightarrow & a \\
\langle b\rangle & \rightarrow & b \\
\langle c\rangle & \rightarrow & c \\
\langle BC\rangle & \rightarrow & BC \\
\langle\langle a\rangle B\rangle & \rightarrow & \langle a\rangle B \\
\langle\langle a\rangle D\rangle & \rightarrow & \langle a\rangle D \\
\langle CSS\rangle & \rightarrow & \langle CS\rangle S \\
\langle CS\rangle & \rightarrow & CS \\
\langle SS\rangle & \rightarrow & SS
\end{array}
$$

6. Wiederherstellen von ϵ.

   Da S nicht ϵ erzeugen kann, müssen wir kein neues Startsymbol S_{neu} hinzufügen.

Bonusaufgabe 4.5:

von
6

1. Zeigen Sie mit Hilfe des Pumping-Lemmas für kontextfreie Sprachen, dass die Sprache $L = \{a^k b^l \mid l = k^2, k > 0\}$ nicht kontextfrei ist.

   Lösung: Wäre L kontextfrei, dann würde das Pumping-Lemma gelten. Sei n die PL-Zahl. Wähle $z = a^n b^{n^2}$. Da $|z| \geq n$ gilt, muss es eine Zerlegung $z = uvwxy$ geben, so dass $|vwx| \leq n$ und $|vx| \geq 1$ gilt.

   Wir machen eine Fallunterscheidung bzgl. der möglichen Formen von vwx:

   - $vwx \in \{a\}^*$.
     In diesem Fall ist auch $v, x \in \{a\}^*$. Sei also $vx = a^j$ für ein $j > 0$, denn $|vx| \geq 1$.
     Dann müsste auch $uv^iwx^iy = a^n a^{j(i-1)} b^{n^2}$ für jedes i in L sein.
     Für $i = 2$ ist dann $z' = a^{n+j} b^{n^2}$ nicht mehr in L, da mit $j > 0$ ist $k^2 = (n+j)^2 > n^2 = l$.
     Widerspruch für diesen Fall.
   - $vwx \in \{b\}^*$.
     In diesem Fall ist auch $v, x \in \{b\}^*$. Sei also $vx = b^j$ für ein $j > 0$, denn $|vx| \geq 1$.
     Dann müsste auch $uv^iwx^iy = a^n b^{n^2} b^{j(i-1)}$ für jedes i in L sein.
     Für $i = 0$ ist dann $z' = a^n b^{n^2-j}$ nicht mehr in L, da mit $k > 0$ ist $l = n^2 - k < n^2 = k^2$.
     Widerspruch für diesen Fall.
   - $vwx \in \{a\}^+\{b\}^+$.
     Wir stellen fest, dass weder v noch x aus $\{a\}^+\{b\}^+$ sein kann, denn dann erhielten wir durch das Hoch-Pumpen ein Wort z', das einen Wechsel von b auf a hätte. Dann wäre aber z' nicht in L.
     Wäre v aus $\{b\}^*$, dann wäre $vwx \in \{b\}^+$ und den Fall haben wir schon behandelt.
     Also muss $v \in \{a\}^*$ und $x \in \{b\}^*$ sein.
     Sei also $v = a^p$ und $x = b^q$. Dann müsste auch $z' = uv^iwx^iy = uvv^{i-1}wxx^{(i-1)}y = (a^p)(i-1)a^n b^{n^2} (b^q)^{(i-1)}$ in L sein.
     Dann ist $k^2 = (p(i-1)+n)^2 = (p+n)^2$ und $l = (n^2 + q(i-1))$
     Wegen $|vx| \geq 1$ gilt $p + q > 0$. Es gibt also drei Fälle:
     (a) $p = 0, q > 0$.
     Dann ist $k^2 = (p(i-1)+n)^2 = n^2$ und $l = (n^2 + q(i-1))$.
     Für $i = 2$ ist dann $k^2 = n^2 < n^2 + q = l$. z' ist also nicht in L. Widerspruch.
     (b) $p > 0, q = 0$.
     Dann ist $k^2 = (p(i-1)+n)^2 = (p+n)^2$ und $l = (n^2 + q(i-1)) = n^2$.
     Für $i = 2$ ist dann $k^2 = (n+q)^2 > n^2 = l$. z' ist also nicht in L. Widerspruch.
     (c) $p > 0, q > 0$. Dann ist $k^2 = (p(i-1)+n)^2 = (p+n)^2$ und $l = (n^2 + q(i-1)) = n^2$.
     Für $i > q + 1$ ist dann $k^2 > l$, denn es gilt:
     $$k^2 = (n+p(i-1))^2 = n^2+2np(i-1)+p^2(i-1)^2 > n^2+p^2(i-1)^2 > n^2+p^2q(i-1) \geq n^2+q(i-1) = l$$
     z' ist also nicht in L. Widerspruch.

     Widerspruch für diesen Fall.

   Da es keine weiteren Zerlegungen von z mit $|vwx| \leq n$ gibt und alle Fälle zum Widerspruch führen, kann L nicht kontextfrei sein.

2. Zeigen Sie, dass es keine rechtslineare Grammatik G mit $L(G) = L$ geben kann, wobei:

$$L := \{w = \{0,1\}^* \mid \; w \text{ enthält genausoviele } 0 \text{ wie } 1\}$$

Lösung: Man kann leicht zeigen, dass L nicht regulär ist. (Wäre L regulär, dann müsste auch $h(L) \cap \{a\}^*\{b\}^* = DUP$ mit dem durch $h(0) = a$ und $h(1) = b$ definierten Homomorphismus regulär sein. DUP ist aber nicht regulär. Widerspruch.)

Gäbe es eine rechts-lineare Grammatik, die L erzeugt, dann gäbe es nach Satz 15.1 einen NFA, der L akzeptiert und L wäre regulär. Widerspruch. Also kann es eine solche Grammatik nicht geben.