# Formale Sprachen und Komplexität, SS 18, Prof. Dr. Volker Heun

## Übungsblatt 5

*Abgabe: bis Fr. 04.06.2018 8 Uhr*

**Institut für Informatik** **Ludwig-Maximilians-Universität München**
Prof. Dr. Volker Heun

**Formale Sprachen und Komplexität**, SS 18
**Übungsblatt 5**

*Abgabe: bis Fr. 04.06.2018 8 Uhr*

Nach Bearbeitung dieses Übungsblattes sollten Sie:

| | Check |
|---|---|
| Mit dem Index der Äquivalenzrelation $R_L$ umgehen können. | |
| Eine kontextfreie Grammatik in Chomsky Normalform überführen können. | |
| Die Konfigurationen eines Kellerautomaten zu einem Eingabewort bestimmen können. | |
| Die akzeptierte Sprache eines Kellerautomaten bestimmen können. | |
| Das Pumping-Lemma für kontextfreie Sprachen kennen. | |
| Mit Hilfe des Pumping-Lemmas für kontextfreie Sprachen kontrapositiv Beweise führen können. | |

Diese Ziele sind wichtige Hinweise für die Klausur!

### Aufgabe 5-1 Äquivalenzrelation $R_L$

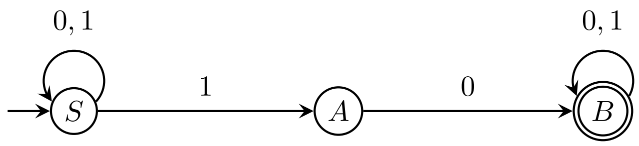
Sei $\Sigma = \{0, 1\}$ und $L = \{w \in \Sigma^* \mid w \text{ enthält das Teilwort } 10\}$.

Zeige, dass $Index(R_L) = 3$ gilt, ohne die Äquivalenzklassen von $R_L$ anzugeben.

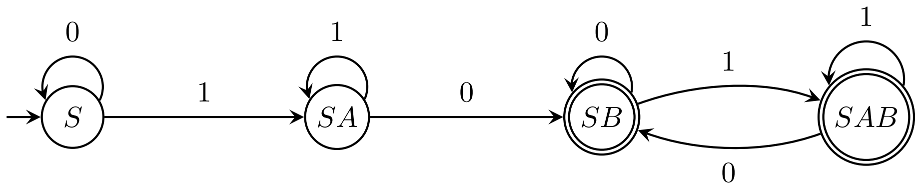
**Lösungsvorschlag:**

**1. Nichtdeterministischen endlichen Automaten** für $L$ angeben
eventuell auf dem Umweg über eine Grammatik oder einen regulären Ausdruck

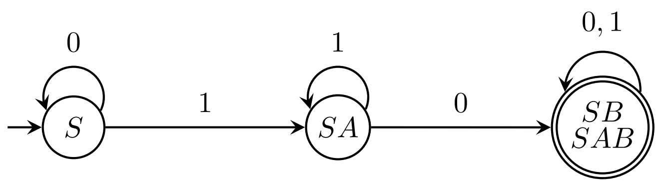
$$L = L(\alpha) \quad \text{für} \quad \alpha = (0|1)^* 10\, (0|1)^*$$




**2. Deterministischen endlichen Automaten** daraus konstruieren




**3. Minimalautomat** daraus konstruieren




$Index(R_L)$ = **Anzahl Zustände des Minimalautomaten** für $L$, hier also 3

**Aufgabe 5-2** schriftlich bearbeiten
**Chomsky-Normalform**

Sei $G = (V, \Sigma, P, S)$ die Grammatik mit $V = \{S, P, Q, R, T, U, X, Y\}$ und $\Sigma = \{a, b, c\}$ und

$$
\begin{array}{llllll}
P = \{ & S \to PQT, & P \to bQ, & Q \to aRY, & & \\
& R \to T, & & T \to U, & & \\
& U \to aU, & U \to bX, & & & \\
& X \to Y, & Y \to X, & Y \to SX & Y \to c & \}
\end{array}
$$

Welche Sprache $L(G)$ dadurch definiert wird, ist für die Aufgabe unwichtig. Konstruieren Sie nach dem Verfahren im Buch eine Grammatik $G'$ in Chomsky-Normalform mit $L(G) = L(G')$.

**Lösungsvorschlag:**

---

**0. $\varepsilon$-Produktionen beseitigen** (falls überhaupt erlaubt)

entfällt hier

---

**1. Kettenproduktionen beseitigen**

1.a) Zyklus $X \to Y \to X$ durch neues $Z$ ersetzen:

| | | | |
|---|---|---|---|
| $S \to PQT$, | $P \to bQ$, | $Q \to aRZ$, | |
| $R \to T$, | | $T \to U$, | |
| $U \to aU$, | $U \to bZ$, | $Z \to SZ$, | $Z \to c$ |

1.b) Kette $R \to T \to U$ von hinten her beseitigen:

in $T \to U$ einsetzen:

| | | | |
|---|---|---|---|
| $S \to PQT$, | $P \to bQ$, | $Q \to aRZ$, | |
| $R \to T$, | | $T \to aU$, | $T \to bZ$, |
| $U \to aU$, | $U \to bZ$, | $Z \to SZ$, | $Z \to c$ |

in $R \to T$ einsetzen:

| | | | |
|---|---|---|---|
| $S \to PQT$, | $P \to bQ$, | $Q \to aRZ$, | |
| $R \to aU$, | $R \to bZ$, | $T \to aU$, | $T \to bZ$, |
| $U \to aU$, | $U \to bZ$, | $Z \to SZ$, | $Z \to c$ |

---

**2. Separieren** (neue Variable für jedes Terminalsymbol)

Bereits separiert für $c$, also nur $A$ für $a$ und $B$ für $b$ erforderlich

| | | | |
|---|---|---|---|
| $S \to PQT$, | $P \to BQ$, | $Q \to ARZ$, | |
| $R \to AU$, | $R \to BZ$, | $T \to AU$, | $T \to BZ$, |
| $U \to AU$, | $U \to BZ$, | $Z \to SZ$, | |
| $A \to a$, | $B \to b$, | | $Z \to c$ |

---

**3. Zerlegen der rechten Seiten mit Länge** $> 2$

| | | | |
|---|---|---|---|
| $S \to PE$, | | $Q \to AF$, | |
| $E \to QT$, | $P \to BQ$, | $F \to RZ$, | |
| $R \to AU$, | $R \to BZ$, | $T \to AU$, | $T \to BZ$, |
| $U \to AU$, | $U \to BZ$, | $Z \to SZ$, | |
| $A \to a$, | $B \to b$, | | $Z \to c$ |

**Aufgabe 5-3** schriftlich bearbeiten
**CYK-Algorithmus**

Sei $G = (V, \Sigma, P, S)$ die Grammatik mit $V = \{S, A, B, C, D, E\}$ und $\Sigma = \{a, b\}$ und

$$
\begin{array}{lllll}
P = \{ & S \to AB, & S \to BC, & S \to BE, & \\
& A \to BA, & & & A \to a, \\
& B \to AD, & B \to EE, & & B \to b, \\
& C \to AB, & & & \\
& D \to BC, & & & \\
& & & & E \to a \quad \}
\end{array}
$$

Bestimmen Sie mit Hilfe des Algorithmus von Cocke, Younger und Kasami (CYK), ob folgende Wörter zu $L(G)$ gehören:

a) *abbaba*

**Lösungsvorschlag:**

| $a$ | $b$ | $b$ | $a$ | $b$ | $a$ |
|---|---|---|---|---|---|
| A,E | B | B | A,E | B | A,E |
| S,C | — | A,S | S,C | A,S | |
| — | A | S,D,C | — | | |
| — | S,D,C | — | | | |
| B | — | | | | |
| **S**,A | | | | | |

b) *aaaab*

**Lösungsvorschlag:**

| $a$ | $a$ | $a$ | $a$ | $b$ |
|---|---|---|---|---|
| A,E | A,E | A,E | A,E | B |
| B | B | B | S,C | |
| S,C,A | S,C,A | — | | |
| — | S,D,C | | | |
| B | | | | |

**Aufgabe 5-4** schriftlich bearbeiten
**Kellerautomaten**

Sei $M = (Z, \Sigma, \Gamma, \delta, z, \#)$ mit $Z = \{z, z'\}$ $\Sigma = \{a, b\}$ $\Gamma = \{A, B, \#\}$

| $\delta :$ | | | | |
|---|---|---|---|---|
| | $1: \; z,\, \# \xrightarrow{a} z,\, A\#$ | $4: \; z,\, \# \xrightarrow{b} z,\, B\#$ | $7: \; z,\, A \xrightarrow{\varepsilon} z',\, A$ | $9: \; z',\, A \xrightarrow{\varepsilon} z',\, \varepsilon$ |
| | $2: \; z,\, A \xrightarrow{a} z,\, AA$ | $5: \; z,\, B \xrightarrow{b} z,\, BB$ | $8: \; z,\, B \xrightarrow{\varepsilon} z',\, B$ | $10: \; z',\, B \xrightarrow{\varepsilon} z',\, \varepsilon$ |
| | $3: \; z,\, B \xrightarrow{a} z,\, \varepsilon$ | $6: \; z,\, A \xrightarrow{b} z,\, \varepsilon$ | | $11: \; z',\, \# \xrightarrow{\varepsilon} z',\, \varepsilon$ |

Die Nummern sind nicht Teil des Formalismus. Sie sollen nur erleichtern, in kompakter Form über die Übergänge zu sprechen. Die Spezifikation von $\delta$ verwendet hier
nicht die Langnotation: $\delta(z, c, X) \ni (z', X'_1 \cdots X'_k)$ aus dem Buch,
sondern die Kurznotation: $z,\, X \xrightarrow{c} z',\, X'_1 \cdots X'_k$

Dieser Kellerautomat akzeptiert durch leeren Keller und leeres Wort.

a) Geben Sie für jedes der folgenden Eingabewörter den Baum aller Konfigurationen an, die $M$ mit dem Eingabewort erreicht: $a$, $b$, $ab$, $bba$, $bbaa$.

**Lösungsvorschlag:**

$(z,\, a,\, \#)$

$1 \downarrow$

$(z,\, \varepsilon,\, A\#) \vdash_{7} (z',\, \varepsilon,\, A\#) \vdash_{9} (z',\, \varepsilon,\, \#) \vdash_{11} (z',\, \varepsilon,\, \varepsilon)$ akzeptiert

---

$(z,\, b,\, \#)$

$4 \downarrow$

$(z,\, \varepsilon,\, B\#) \vdash_{8} (z',\, \varepsilon,\, B\#) \vdash_{10} (z',\, \varepsilon,\, \#) \vdash_{11} (z',\, \varepsilon,\, \varepsilon)$ akzeptiert

---

$(z,\, ab,\, \#)$

$1 \downarrow$

$(z,\, b,\, A\#) \vdash_{7} (z',\, b,\, A\#) \vdash_{9} (z',\, b,\, \#) \vdash_{11} (z',\, b,\, \varepsilon)$ Sackgasse

$6 \downarrow$

$(z,\, \varepsilon,\, \#)$ Sackgasse

**Lösungsvorschlag:**

---

$$\begin{array}{lllllll}
(z,\ bba,\ \#) & & & & & & \\
4\,\top & & & & & & \\
(z,\ ba,\ B\#) & \vdash_{8} & (z',\ ba,\ B\#) & \vdash_{10} & (z',\ ba,\ \#) & \vdash_{11} & (z',\ ba,\ \varepsilon) \\
 & & & & & & \text{Sackgasse} \\
5\,\top & & & & & & \\
(z,\ a,\ BB\#) & \vdash_{8} & (z',\ a,\ BB\#) & \vdash_{10}^{+} & (z',\ a,\ \#) & \vdash_{11} & (z',\ a,\ \varepsilon) \\
 & & & & & & \text{Sackgasse} \\
3\,\top & & & & & & \\
(z,\ \varepsilon,\ B\#) & \vdash_{8} & (z',\ \varepsilon,\ B\#) & \vdash_{10} & (z',\ \varepsilon,\ \#) & \vdash_{11} & (z',\ \varepsilon,\ \varepsilon) \\
 & & & & & & \text{akzeptiert}
\end{array}$$

---

$$\begin{array}{lllllll}
(z,\ bbaa,\ \#) & & & & & & \\
4\,\top & & & & & & \\
(z,\ baa,\ B\#) & \vdash_{8} & (z',\ baa,\ B\#) & \vdash_{10} & (z',\ baa,\ \#) & \vdash_{11} & (z',\ baa,\ \varepsilon) \\
 & & & & & & \text{Sackgasse} \\
5\,\top & & & & & & \\
(z,\ aa,\ BB\#) & \vdash_{8} & (z',\ aa,\ BB\#) & \vdash_{10}^{+} & (z',\ aa,\ \#) & \vdash_{11} & (z',\ aa,\ \varepsilon) \\
 & & & & & & \text{Sackgasse} \\
3\,\top & & & & & & \\
(z,\ a,\ B\#) & \vdash_{8} & (z',\ a,\ B\#) & \vdash_{10} & (z',\ a,\ \#) & \vdash_{11} & (z',\ a,\ \varepsilon) \\
 & & & & & & \text{Sackgasse} \\
3\,\top & & & & & & \\
(z,\ \varepsilon,\ \#) & & & & & & \\
\text{Sackgasse} & & & & & &
\end{array}$$

b) Geben Sie die Sprache $N(M)$ an, die der Kellerautomat $M$ akzeptiert.

**Lösungsvorschlag:**

Der Automat speichert im Keller
den „Überschuss“ der $a$'s bzw. der $b$'s im bisher gelesenen Teilwort.

Hat es gleich viele $a$'s wie $b$'s, wird das durch Kellerinhalt $\#$ repräsentiert.
Hat es zum Beispiel zwei $a$'s „zu viel“, wird das durch Kellerinhalt $AA\#$ repräsentiert.
Drei $b$'s „zu viel“ werden durch Kellerinhalt $BBB\#$ repräsentiert.

Von einer Konfiguration mit Zustand $z$
kann eine Konfiguration mit leerem Keller nur erreicht werden,
wenn ein Zustandswechsel von $z$ nach $z'$ stattfindet.

Zustandswechsel von $z$ nach $z'$ sind nur von Konfigurationen aus möglich,
in denen der Keller mehr enthält als das Kellerendezeichen,
also nur wenn das bisherige Wort einen Überschuss von $a$'s oder von $b$'s hat.

Der Automat akzeptiert die Menge aller Wörter über $\{a, b\}$,
in denen ungleich viele $a$'s wie $b$'s vorkommen.

Für $w \in \Sigma^*$ sei $|w|_a$ die Anzahl der $a$'s und $|w|_b$ die Anzahl der $b$'s im Wort $w$.

$$N(M) \quad = \quad \left\{ \ w \in \{a, b\}^* \ \middle| \ |w|_a \neq |w|_b \ \right\}$$

## Aufgabe 5-5 Pumping-Lemma (kontextfrei)

Beweisen oder widerlegen Sie: folgende formale Sprachen sind vom Typ 2 (kontextfrei).

**Lösungsvorschlag:**

**Pumping-Lemma Typ 3 (regulär)**

Sei $L \subseteq \Sigma^*$ eine formale Sprache.
Wenn $L$ vom Typ 3 (regulär) ist,
dann
$\exists n \in \mathbb{N}$ (Pumping-Lemma-Zahl von $L$) so dass
$\forall z \in L$ mit $|z| \geq n$ gilt
$\exists u,v,w \in \Sigma^*$ mit $z = uvw$ (Zerlegung von $z$) so dass

1. $1 \leq |v|$
2. $|uv| \leq n$
3. $\forall i \in \mathbb{N}$ gilt $uv^iw \in L$

**Pumping-Lemma Typ 2 (kontextfrei)**

Sei $L \subseteq \Sigma^*$ eine formale Sprache.
Wenn $L$ vom Typ 2 (kontextfrei) ist,
dann
$\exists n \in \mathbb{N}$ (Pumping-Lemma-Zahl von $L$) so dass
$\forall z \in L$ mit $|z| \geq n$ gilt
$\exists u,v,w,x,y \in \Sigma^*$ mit $z = uvwxy$ (Zerlegung von $z$) so dass

1. $1 \leq |vx|$
2. $|vwx| \leq n$
3. $\forall i \in \mathbb{N}$ gilt $uv^iwx^iy \in L$

**Vergleich:**
Bis zur Zerlegung sind die beiden Lemmata völlig analog aufgebaut.
Erst mit der Zerlegung beginnen die Unterschiede.

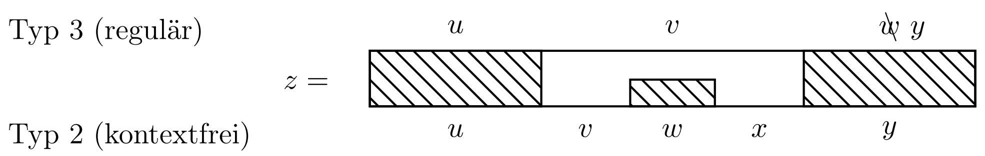
**Lösungsvorschlag:**

**Zerlegung im Pumping-Lemma Typ 3 (regulär)**

In jedem hinreichend langen Wort $z$ der Sprache wird ein kritisches Teilwort $v$ identifiziert, mit dem man $z$ „aufpumpen" kann. Die Zerlegung bezeichnet alles links von $v$ als $u$ und alles rechts von $v$ als $w$. Aber den uninteressanten rechten Teil $w$ können wir genausogut $y$ nennen (dann wird $w$ wieder frei).

Die drei Forderungen an die Zerlegung garantieren, dass das kritische Teilwort:

1. nichttrivial ist.
2. a) höchstens die Länge $n$ hat
   b) innerhalb der ersten $n$ Zeichen von $z$ liegt.
3. zum „aufpumpen" von $z$ verwendet werden kann.




**Zerlegung im Pumping-Lemma Typ 2 (kontextfrei)**

Das kritische Teilwort ist in zwei Teilteilwörter aufgeteilt, die $v$ und $x$ genannt werden. Das kritische Teilwort selbst ist also $vx$. Aber seine beiden Teile können voneinander getrennt sein durch ein uninteressantes Zwischenwort $w$.

Die drei Forderungen an die Zerlegung garantieren, dass das kritische Teilwort:

1. nichttrivial ist. — wie vorher
2. a) höchstens die Länge $n$ hat — wie vorher
   b) durch das Zwischenwort $w$ nicht zu weit auseinandergedrängt wird.
   Der Anfang seines ersten und das Ende seines zweiten Teilteilworts
   können höchstens die Entfernung $n$ haben.
   Das kritische Teilwort muss aber nicht mehr am Anfang von $z$ liegen. — anders
3. zum „aufpumpen" von $z$ verwendet werden kann. — wie vorher
   Das schaut nur technisch etwas anders aus,
   weil dabei das Zwischenwort $w$ unberührt bleibt.

Dass $vx$ das kritische Teilwort ist, äußert sich auch darin,
dass in Widerspruchsbeweisen mit dem Pumpinglemma Typ 2
fast immer Fallunterscheidungen nach der Gestalt von $vx$ vorkommen.

a) $L = \{\ a^i b^k c^j \mid i, j, k \in \mathbb{N} \text{ und } 0 < i < k \text{ und } 0 < j < k\ \}$.

Jedes Wort der Sprache enthält also mindestens ein $a$ am Anfang, mindestens ein $c$ am Ende, und die Anzahl von $a$'s und von $c$'s ist voneinander unabhängig. Die Anzahl der $b$'s dazwischen ist größer als jede der anderen beiden Anzahlen.

**Lösungsvorschlag:**

**Behauptung:** $L$ ist nicht vom Typ 2 (kontextfrei).

**Beweis:** Widerspruchsbeweis mit dem Pumping-Lemma

Annahme: $L$ sei vom Typ 2 (kontextfrei).
Sei $n$ eine Pumping-Lemma-Zahl von $L$.
Betrachte $z = a^n b^{n+1} c^n \in L$. Es gilt $|z| = 3n + 1 \geq n$.

Sei $z = uvwxy$ eine Zerlegung mit
$1 \leq |vx| \qquad |vwx| \leq n \qquad \forall i \in \mathbb{N}$ gilt $uv^iwx^iy \in L$

Für diese Zerlegung gilt $uvwxy = a^n b^{n+1} c^n$ und $|vwx| \leq n$.
Im Teilwort $vwx$ können also $a$ und $c$ nicht beide vorkommen.
Es gibt nur folgende Fälle:

**Fall** $vx = a^m$ mit $1 \leq m \leq n$:
Dann ist $uv^2wx^2y = a^{n+m}\, b^{n+1}\, c^n \notin L$ da $n + m \not< n + 1$ Widerspruch.

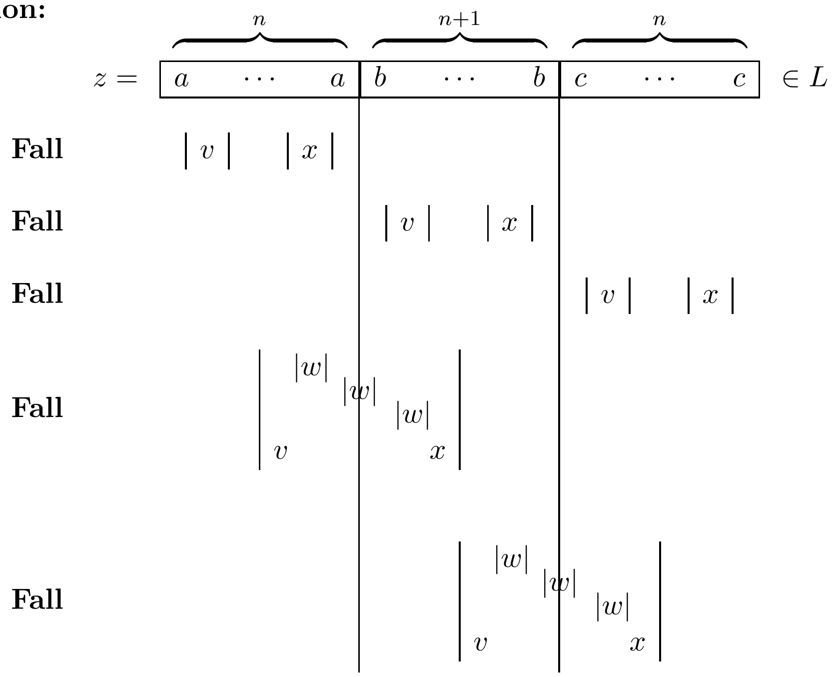
**Fall** $vx = b^m$ mit $1 \leq m \leq n$:
Dann ist $uv^0wx^0y = a^n\, b^{n+1-m}\, c^n \notin L$ da $n \not< n + 1 - m$ Widerspruch.

**Fall** $vx = c^m$ mit $1 \leq m \leq n$:
Dann ist $uv^2wx^2y = a^n\, b^{n+1}\, c^{n+m} \notin L$ da $n + m \not< n + 1$ Widerspruch.

**Fall** $vx = a^{m'} b^m$ mit $1 \leq m' < n$ und $1 \leq m < n$:
Dann ist $uv^0wx^0y = a^{n-m'}\, b^{n+1-m}\, c^n \notin L$ da $n \not< n + 1 - m$ Widerspruch.

**Fall** $vx = b^m c^{m'}$ mit $1 \leq m < n$ und $1 \leq m' < n$:
Dann ist $uv^0wx^0y = a^n\, b^{n+1-m}\, c^{n-m'} \notin L$ da $n \not< n + 1 - m$ Widerspruch.

**Illustration:**




In den letzten beiden Fällen hängt der Widerspruch nicht davon ab, wo $w$ genau liegt. Es reicht, dass $v$ mit dem einen Zeichen beginnt und $x$ mit dem anderen endet.

b) $L' = \{\ a^i b^k c^j \mid i, j, k \in \mathbb{N}$ und $0 < i$ und $0 < j$ und $i + j < k\ \}$.

Jedes Wort der Sprache enthält also mindestens ein $a$ am Anfang, mindestens ein $c$ am Ende, und die Anzahl von $a$'s und von $c$'s ist voneinander unabhängig. Die Anzahl der $b$'s dazwischen ist größer als die Summe der anderen beiden Anzahlen.

**Lösungsvorschlag:**

**Behauptung:** $L$ ist vom Typ 2 (kontextfrei).

**Für den Beweis dieser Behauptung nützt das Pumping-Lemma nichts!**
Wir zeigen, dass es eine Grammatik $G$ vom Typ 2 (kontextfrei) gibt mit $L(G) = L$.

**Beweis:** Sei $G = (V, \Sigma, P, S)$ mit $V = \{S, A, B, C\}$ und

$$
\begin{array}{lllll}
P = \{ & S \to ABC, & & & \\
 & & A \to ab, & B \to b, & C \to bc, \\
 & & A \to aAb, & B \to bB, & C \to bCc \quad \}
\end{array}
$$

Diese Produktionen wirken so:

$A$ erzeugt genau die Terminalwörter der Gestalt $a^i b^i$ mit $i \geq 1$.
$B$ erzeugt genau die Terminalwörter der Gestalt $b^\ell$ mit $\ell \geq 1$.
$C$ erzeugt genau die Terminalwörter der Gestalt $b^j c^j$ mit $j \geq 1$.

Wenn die Startproduktion $S \to AC$ wäre,
entstünden aus $S$ genau die Terminalwörter der Gestalt $a^i b^{i+j} c^j$ also $i + j = k$.

Mit der Startproduktion $S \to ABC$
erzeugt $S$ genau die Terminalwörter der Gestalt $a^i b^{i+\ell+j} c^j$ also $i + j < k = i + j + \ell$.