Karlsruher Institut für Technologie
Institut für Theoretische Informatik

Prof. Dr. Peter Sanders
L. Hübschle-Schneider, T. Maier

5. Übungsblatt zu Theoretische Grundlagen der Informatik im WS 2015/16

http://algo2.iti.kit.edu/TGI2015.php
{sanders,huebschle,t.maier}@kit.edu

Musterlösungen

Aufgabe 1 (*Pumpinglemma für kontextfreie Sprachen,* $2+4+4$ *Punkte*)

Sie die folgenden Sprachen kontextfrei? Zeigen Sie durch Angabe einer Grammatik oder widerlegen Sie mithilfe des Pumpinglemmas für kontextfreie Sparchen.

a) $L_1 = \{a^n b^m c^n d^m \mid n, m \in \mathbb{N}\}$ über dem Alphabet $\Sigma_1 = \{a, b, c, d\}$

b) $L_2 = \{b_i \# b_{i+1} \mid i \in \mathbb{N},\ b_n \text{ ist die Binärzahlrepresentation von } n \in \mathbb{N}\}$ und $L_2' = \{b_i \# b_{i+1}^R \mid i \in \mathbb{N}\}$, beide über $\Sigma_2 = \{0, 1, \#\}$

c) $L_3 = \{ww^R w \mid w \in \Sigma_3^*\}$ über $\Sigma_3 = \{a, b\}$

Musterlösung:

a) Angenommen, L_1 sei regulär. Sei n der Wert ab dem das Pumpinglemma gilt. Wähe $z = a^n b^n c^n d^n \in L_1$. Betrachte eine beliebige Zerlegung $z = uvwxy$, für die $|vx| \geq 1$ und $|vwx| \leq n$ gilt. Betrachte die folgenden Fälle:

- v oder x enthalten mindestens zwei verschiedene Zeichen. Enthalte v sowohl as als auch bs: $v = a^r b^s,\ r, s \in \mathbb{N}$. Jedoch liegt $z' = uv^2wx^2z = ua^r b^s a^r b^s wx^2 y$ nicht in L_1, Widerspruch. Die anderen Fälle verlaufen analog ($v = b^r c^s$, $v = c^r d^s$, $x = a^r b^s$, $x = b^r c^s$, $x = c^r d^s$, und Kombinationen davon).
- Sowohl v als auch x enthalten jeweils nur ein (möglicherweise wiederholtes) Zeichen. ObdA sei $v = a^r,\ r \in \mathbb{N}$ (die Fälle $v = b^r$, $v = c^r$ und $v = d^r$ verlaufen analog). Da $|vwx| \leq n$ folgt daraus $x = a^s$ oder $x = b^s,\ s \in \mathbb{N}$. Dann liegt jedoch $z' = uv^2wx^2y$ nicht in L_1, da es entweder mehr as als cs oder mehr bs als ds (oder beides) enthält.

Daher kann L_1 nicht kontextfrei sein.

b) Angenommen, L_2 sei regulär. Sei n der Wert ab dem das Pumpinglemma gilt. Wähle $z = (10)^n 01^n \#(10)^n 10^n \in L_2$. Betrachte eine beliebige Zerlegung $z = uvwxy$, für die $|vx| \geq 1$ und $|vwx| \leq n$ gilt. Dann gilt $z' = uv^0wx^0y = uwy \in L_2$, d.h., weder v noch x enthalten das Zeichen $\#$. Betrachte nun die folgenden Fälle:

- Wenn vwx nur Zeichen aus der ersten Zahl (links des $\#$) enthält, wird die Zahl, die durch diese Bits repräsentiert wird, durch pumpen mit $i > 1$ größer als die, die durch die Bits auf der rechten Seite repräsentierte. Daher kann uv^2wx^2y nicht in L_2 liegen.
- Wenn vwx nur Zeichen aus der zweiten Zahl enthält (rechts des $\#$), wird die davon repräsentierte Zahl durch pumpen mit $i > 1$ viel größer als die durch die Bits auf der linken Seite repräsentierte Zahl. Die Differenz wird dadurch größer als 1 und $uv^2wx^2y \notin L_2$.
- Wenn vwx Zeichen aus beiden Zahlen enthält, dann muss es aufgrund der Längenbeschränkung für den Mittelteil die Form $1^s\#(10)^t$ ($s, t \in \mathbb{N}$) oder $1^s\#(10)^t 1$ ($s \in \mathbb{N}, t \in \mathbb{N}_0$) haben. Wir führen wiederum alle Fälle zum Widerspruch:

– Wenn v nur Bits von der linken Seite des # und x nur Bits von dessen rechter seite enthält, bleibt die relative Länge von linker und rechter Seite nach dem pumpen gleich, die Differenz zwischen den Zahlen beträgt aber nicht mehr genau eins.
– Ansonsten ändert sich die relative Länge von linker und rechter Seite. Wie wir bereits oben gesehen haben, führt das zum Widerspruch.

Daher kann L_2 nicht kontextfrei sein. L_2' hingegen ist kontextfrei, die Grammatik ist recht einfach: $G = (\{S, A\}, \Sigma_2, P, S), P = \{\, S \longrightarrow 0S0 \mid 1S1 \mid 0A1, \quad A \longrightarrow 1A0 \mid \# \,\}$.

c) Angenommen, L_3 sei regulär. Sei n der Wert ab dem das Pumpinglemma gilt. Wähle $z = (a^k b^k)(b^k a^k)(a^k b^k) = a^k b^{2k} a^{2k} b^k$. Betrachte eine beliebige Zerlegung $z = uvwxy$, für die $|vx| \geq 1$ und $|vwx| \leq n$ gilt. Wir müssen vier Fälle betrachten: Die Teilworte v und x können Zeichen des

1) 1. a-Blocks	2) 1. b-Blocks	3) 2. a-Blocks	4) 2. b-Blocks

enthalten. Wir betrachten diese Fälle im Folgenden getrennt. Da $|vwx| \leq n$, können v und x keine

1) as des 2. a-Blocks	2) bs des 2. b-Blocks	3) as des 1. a-Blocks	4) bs des 1. b-Blocks

enthalten. Dann muss $z' = uv^0wx^0y = uwy$ die Form

1) $a^{i_1}b^{i_2}a^{2k}b^k$ mit $i_1 < k, i_2 \leq 2k$	2) $a^{i_1}b^{i_2}a^{i_3}b^k$ mit $i_1 \leq k, i_2 < 2k, i_3 \leq 2k$	3) $a^k b^{i_1} a^{i_2} b^{i_3}$ mit $i_1 \leq 2k, i_2 < 2k, i_3 \leq k$	4) $a^k b^{2k} a^{i_1} b^{i_2}$ mit $i_1 \leq 2k, i_2 < k$

haben. Wenn $z' \in L_3$, muss es aber auch die Form $z' = \alpha\alpha^R\alpha$ haben. Aufgrund seiner oben beschriebenen Form und seiner Länge von mindestens $5n$, muss das

1) erste	2) letzte	3) erste	4) letzte

α mit einem Block von

1) $i_1 < k$ as beginnen	2) k as enden	3) k as beginnen	4) $i_2 < k$ bs enden

gefolgt von einer Anzahl des jeweils anderen Zeichens. Daher muss

1) $\alpha^R\alpha$	2) $\alpha\alpha^R$	3) $\alpha^R\alpha$	4) $\alpha\alpha^R$

einen Block von

1) $\leq 2i_1 < 2k$ as	2) $2k$ bs	3) $2k$ as	4) $\leq 2i_2 < 2k$ bs

enthalten. Da uwy aber einen Block von

1) $2k$ as	2) $i_2 < 2k$ bs	3) $i_2 < 2k$ as	4) $2k$ bs

enthält, führt das zum Widerspruch.

Aufgabe 2 (*CYK-Algorithmus,* $4 + 4$ *Punkte*)

Verwenden Sie die in der Vorlesung vorgestellten Arbeitsschritte, um die folgenden Grammatiken in eine Form zu bringen, so dass der CYK-Algorithmus verwendet werden kann. Führen Sie danach den CYK-Algorithmus durch, um das Wortproblem der gegebenen Worte zu lösen (w_x).

$G_x = (\{S, E, F, G\}, \{a, b, c\}, P_x, S)$

a) $w_a = adabbbcc$

$$P_a = \{S \Rightarrow EF,\quad E \Rightarrow H \mid aEb \mid \varepsilon,\quad F \Rightarrow bFc \mid \varepsilon,\quad H \Rightarrow dH \mid E\}$$

b) $w_{b1} = ababba$
$w_{b2} = aaabbb$

$$P_b = \{S \Rightarrow aSbS \mid bSaS \mid \varepsilon\}$$

Musterlösung:

a)

- Entfernen von ε Übergängen (jedes Nichtterminal kann auf ε abgebildet werden!): Das Nichtterminal S bildet weiterhin auf ε ab. Dies ist kein Problem, da ε in der Sprache liegt, und S in keiner Produktion erzeugt werden kann.

$$\begin{aligned} P'_a = \{ & S \Rightarrow \varepsilon \mid E \mid F \mid EF, \\ & E \Rightarrow H \mid ab \mid aEb, \\ & F \Rightarrow bc \mid bFc, \\ & H \Rightarrow d \mid dH \mid E\} \end{aligned}$$

- Auflösen zyklischer Einheitsabbildungen ($E \leftrightarrow H$):

$$\begin{aligned} P''_a = \{ & S \Rightarrow \varepsilon \mid E \mid F \mid EF, \\ & E \Rightarrow ab \mid aEb \mid d \mid dE, \\ & F \Rightarrow bc \mid bFc\} \end{aligned}$$

- Eliminieren nicht-zyklischer Einheitsabbildungen:

$$\begin{aligned} P'''_a = \{ & S \Rightarrow \varepsilon \mid ab \mid aEb \mid d \mid dE \mid bc \mid bFc \mid EF, \\ & E \Rightarrow ab \mid aEb \mid d \mid dE, \\ & F \Rightarrow bc \mid bFc\} \end{aligned}$$

- Verkürzen zu langer rechter Seiten und entfernen unerlaubter Terminalsymbole:

$$\begin{aligned} P''''_a = \{ & S \Rightarrow \varepsilon \mid AB \mid AE_b \mid d \mid DE \mid BC \mid BF_c \mid EF, \\ & E \Rightarrow AB \mid AE_b \mid d \mid DE, \\ & F \Rightarrow BC \mid BF_c, \\ & E_b \Rightarrow EB, \\ & F_c \Rightarrow FC, \\ & A \Rightarrow a, \quad B \Rightarrow b, \quad C \Rightarrow c, \quad D \Rightarrow d\} \end{aligned}$$

w_a:

a	d	a	b	b	b	c	c
A	D, S, E	A	B	B	B	C	C
-	-	S, E	-	-	S, F	-	
-	S, E	E_b	-	-	F_c		
-	E_b	-	-	S, F			
S, E	-	-	-				
-	-	S					
-	S						
-							

Das Wort w_a liegt nicht in der Sprache, dies ist daran erkennbar, dass kein Startsymbol in der untersten Zelle der Tabelle entsteht.

b)

- Entfernen von ε Übergängen: Da das Startsymbol S auf ε abbildet und auch durch andere Produktionen selbst erreicht wird, muss ein neues Startsymbol S' eingeführt werden.

$$\begin{aligned} P'_b = \{ & S' \Rightarrow S \mid \varepsilon, \\ & S \Rightarrow aSbS \mid aSb \mid abS \mid ab \mid bSaS \mid bSa \mid baS \mid ba\} \end{aligned}$$

- Auflösen zyklischer Einheitsabbildungen: Es gibt keine zyklischen Einheitsabbildungen
- Eliminieren nicht-zyklischer Einheitsabbildungen:

$$\begin{aligned} P''_b = \{ & S' \Rightarrow aSbS \mid aSb \mid abS \mid ab \mid bSaS \mid bSa \mid baS \mid ba \mid \varepsilon, \\ & S \Rightarrow aSbS \mid aSb \mid abS \mid ab \mid bSaS \mid bSa \mid baS \mid ba\} \end{aligned}$$

- Verkürzen zu langer rechter Seiten und entfernen unerlaubter Terminalsymbole: In dieser Grammatik erhält man durch scharfes hinschauen deutlich einfachere Aufteilungen in Nichtterminale.

$$\begin{aligned} P''_b = \{ & S' \Rightarrow U_a U_b \mid U_b U_a \mid \varepsilon, \\ & S \Rightarrow U_a U_b \mid U_b U_a, \\ & U_a \Rightarrow AS \mid a, \\ & U_b \Rightarrow BS \mid b, \\ & A \Rightarrow a \quad B \Rightarrow b\} \end{aligned}$$

Da die Grammatiken von jedem Studenten unterschiedlich sein können, sind auch die Tabellen individuell unterschiedlich. **Tipp:** Die Vorkommen der Startsymbole in der Tabelle sind unabhängig von der Grammatik.

w_{b1}:

a	b	a	b	b	a
U_a, A	U_b, B	U_a, A	U_b, B	U_b, B	U_a, A
S, S'	S, S'	S, S'	-	S, S'	
U_a	U_b	-	U_b		
S, S'	-	S, S'			
-	U_b				
S, S'					

w_{b2}:

a	a	a	b	b	b
U_a, A	U_a, A	U_a, A	U_b, B	U_b, B	U_b, B
-	-	S, S'	-	-	
-	U_a	-	-		
-	S, S'	-			
U_a	-				
S, S'					

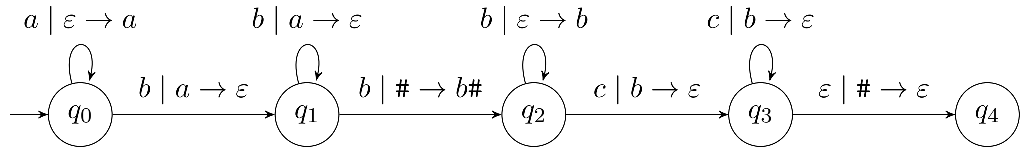
Die Worte w_{b1} und w_{b2} liegen beide in der Sprache, dies ist daran erkennbar, dass S' in der untersten Zelle der Tabelle steht.

Aufgabe 3 (*Kellerautomaten Konstruktion, 2 Punkte*)

Erstellen Sie einen Kellerautomaten für die folgende Sprache.

$L = \{a^i b^{i+j} c^j \mid i, j \in \mathbb{N}\}$

Musterlösung:



Beachte: ein Kellerautomat akzeptiert genau dann, wenn nach der Eingabe des gesammten Wortes ein leerer Stack erreicht werden kann.