TECHNISCHE UNIVERSITÄT DRESDEN

FORMALE SYSTEME

ÜBUNG 7

Eric Kunze
eric.kunze@tu-dresden.de

TU Dresden, 3. Dezember 2021

letzte Änderung:
02.12.2021, 22:01

CORONA-UMFRAGE

Wie stehst du zu Online-Übungen?

- Ich bin für Online-Übungen.
  nur online
- Ich kann mit Online-Übungen leben.
  beides okay, bevorzugt online
- Ich kann mit Präsenzübungen leben.
  beides okay, bevorzugt Präsenz
- Ich bin für Präsenzübungen.
  nur Präsenz

`https://tudvote.tu-dresden.de/88314`

Aufgabe 1:
Kontextfrei? Pumping-Lemma?

PUMPEN FÜR KONTEXTFREIE SPRACHEN

Grundidee beim regulären Pumping-Lemma:
endliche viele Zustände $\rightsquigarrow$ Schleifen für lange Wörter notwendig $\rightsquigarrow$ Schleifen mehrfach benutzbar

PUMPEN FÜR KONTEXTFREIE SPRACHEN

Grundidee beim regulären Pumping-Lemma:
endliche viele Zustände $\rightsquigarrow$ Schleifen für lange Wörter notwendig $\rightsquigarrow$ Schleifen mehrfach benutzbar

Grundidee beim kontextfreien Pumping-Lemma:

- Grammatiken haben nur endlich viele Variablen.
- Beim Generieren langer Wörter muss eine Variable zu etwas expandiert werden, das diese Variable nochmal enthält:
  $$\mathrm{S} \Rightarrow \ldots \Rightarrow u\,\underline{\mathrm{A}}\,y \Rightarrow u\,\underline{z}\,y \Rightarrow \ldots \Rightarrow u\,\underline{v\mathrm{A}x}\,y \Rightarrow \ldots \Rightarrow u\,\underline{vwx}\,y$$
- Diese Schleife kann beliebig oft durchlaufen werden.

Satz (Pumping Lemma): Für jede kontextfreie Sprache $\mathbf{L}$
gibt es eine Zahl $n \geq 0$, so dass gilt:
für jedes Wort $z \in \mathbf{L}$ mit $|z| \geq n$
gibt es eine Zerlegung $z = uvwxy$ mit $|vx| \geq 1$ und $|vwx| \leq n$, so dass:
für jede Zahl $k \geq 0$ gilt: $uv^k wx^k y \in \mathbf{L}$

AUFGABE 1

Welche der folgenden Sprachen L_i ist kontextfrei? Zur Begründung Ihrer Antwort sollten Sie das Pumping-Lemma für kontextfreie Sprachen verwenden oder eine entsprechende kontextfreie Grammatik angeben.

a) $L_1 = \{a^n b^n c^n d^n \in \{a, b, c, d\}^* \mid n \geq 1\}$

b) $L_2 = \{a^m b^n c^p d^q \in \{a, b, c, d\}^* \mid m, n, p, q \geq 1$ und $m + n = p + q\}$

Lösung

a) L_1 ist nicht kontextfrei. Angenommen L_1 wäre kontexfrei, dann gilt das Pumping-Lemma mit einer Zahl $n \geq 0$. Wir betrachten das Wort $w = a^n b^n c^n d^n \in L_1$ und eine Zerlegung dessen in $w = uvwxy$ mit $|vx| \geq 1$ und $|vwx| \leq n$.

- ▷ Fall 1: $vwx \in \{a, b\}^+$, d.h. vwx ist ein Teilwort von $a^n b^n$. Dann enthält uv^2wx^2y mehr a oder b als c oder d. Somit ist $uv^2wx^2y \notin L_1$ im Widerspruch zum Pumping-Lemma.
- ▷ Fall 2: $vwx \in \{b, c\}^+$, d.h. vwx ist ein Teilwort von $b^n c^n$. Analog zu Fall 1.
- ▷ Fall 3: $vwx \in \{c, d\}^+$, d.h. vwx ist ein Teilwort von $c^n d^n$. Analog zu Fall 1.

b) L_2 ist kontextfrei. Betrachte dazu die Grammatik $G = \langle V, \Sigma, P, S \rangle$ mit $V = \{S, S_1, S_2, S_3, S_4\}$ und

$$P = \left\{ \begin{array}{lcl} S & \to & aS_1d \\ S_1 & \to & aS_1d \mid aS_2c \mid bS_3d \mid bS_4c \\ S_2 & \to & aS_2c \mid bS_4c \\ S_3 & \to & bS_3c \mid bS_4c \\ S_4 & \to & bS_4c \mid \varepsilon \end{array} \right\}$$

Aufgabe 2:
Abschlusseigenschaften

ABSCHLUSS FÜR KONTEXTFREIE SPRACHEN

Satz: Wenn $\mathbf{L}$, $\mathbf{L_1}$ und $\mathbf{L_2}$ kontextfreie Sprachen sind, dann beschreiben auch die folgenden Ausdrücke kontextfreie Sprachen:

(1) $\mathbf{L_1} \cup \mathbf{L_2}$ (Abschluss unter Vereinigung)

(2) $\mathbf{L_1} \circ \mathbf{L_2}$ (Abschluss unter Konkatenation)

(3) $\mathbf{L}^*$ (Abschluss unter Kleene-Stern)

Aber:

Satz: Es gibt kontextfreie Sprachen $\mathbf{L}$, $\mathbf{L_1}$ und $\mathbf{L_2}$, so dass die folgenden Ausdrücke keine kontextfreien Sprachen sind:

(1) $\mathbf{L_1} \cap \mathbf{L_2}$ (Nichtabschluss unter Schnitt)

(2) $\overline{\mathbf{L}}$ (Nichtabschluss unter Komplement)

AUFGABE 2

Beweisen Sie mithilfe der Abschlußeigenschaften für kontextfreie Sprachen, dass die nachfolgende Sprache

$$L_0 = \{a^i b^j c^k \mid i = j \text{ oder } j = k \text{ mit } i, j, k \geq 1\}$$

kontextfrei ist und geben dann eine kontextfreie Grammatik G_0 für L_0 an mit $L_0 = L(G_0)$. Geben Sie Ableitungen für die Wörter abc und $abbcc$ an.

Lösung

$$
\begin{aligned}
L_0 &= \left\{a^i b^j c^k : i = j \text{ oder } j = k \text{ mit } i, j, k \geq 1\right\} \\
&= \left\{a^i b^j c^k : i = j \text{ mit } i, j, k \geq 1\right\} \cup \left\{a^i b^j c^k : j = k \text{ mit } i, j, k \geq 1\right\} \\
&= \left\{a^i b^i c^k : i, k \geq 1\right\} \cup \left\{a^i b^j c^j : i, j \geq 1\right\} \\
&= \underbrace{\left\{a^i b^i : i \geq 1\right\}}_{\text{kontextfrei}} \underbrace{\{c\}^*}_{\text{regulär}} \cup \underbrace{\{a\}^*}_{\text{regulär}} \underbrace{\left\{b^j c^j : j \geq 1\right\}}_{\text{kontextfrei}}
\end{aligned}
$$

Lösung

$$
\begin{aligned}
L_0 &= \{a^i b^j c^k : i = j \text{ oder } j = k \text{ mit } i,j,k \geq 1\} \\
&= \{a^i b^j c^k : i = j \text{ mit } i,j,k \geq 1\} \cup \{a^i b^j c^k : j = k \text{ mit } i,j,k \geq 1\} \\
&= \{a^i b^i c^k : i,k \geq 1\} \cup \{a^i b^j c^j : i,j \geq 1\} \\
&= \underbrace{\{a^i b^i : i \geq 1\}}_{\text{kontextfrei}} \underbrace{\{c\}^*}_{\text{regulär}} \cup \underbrace{\{a\}^*}_{\text{regulär}} \underbrace{\{b^j c^j : j \geq 1\}}_{\text{kontextfrei}}
\end{aligned}
$$

Grammatik $G_0 = \langle V, \Sigma, P, S \rangle$ mit $V = \{S, S_a, S_c, S_{ab}, S_{bc}\}$ und

$$
P = \left\{
\begin{array}{lclclcl}
S & \rightarrow & S_{ab}S_c \mid S_a S_{bc} & & & & \\
S_a & \rightarrow & aS_a \mid a & & S_{ab} & \rightarrow & aS_{ab}b \mid ab \\
S_c & \rightarrow & cS_c \mid c & & S_{bc} & \rightarrow & bS_{bc}c \mid bc
\end{array}
\right\}
$$

Lösung

$$
\begin{aligned}
L_0 &= \{a^i b^j c^k : i = j \text{ oder } j = k \text{ mit } i,j,k \geq 1\} \\
&= \{a^i b^j c^k : i = j \text{ mit } i,j,k \geq 1\} \cup \{a^i b^j c^k : j = k \text{ mit } i,j,k \geq 1\} \\
&= \{a^i b^i c^k : i,k \geq 1\} \cup \{a^i b^j c^j : i,j \geq 1\} \\
&= \underbrace{\{a^i b^i : i \geq 1\}}_{\text{kontextfrei}} \underbrace{\{c\}^*}_{\text{regulär}} \cup \underbrace{\{a\}^*}_{\text{regulär}} \underbrace{\{b^j c^j : j \geq 1\}}_{\text{kontextfrei}}
\end{aligned}
$$

Grammatik $G_0 = \langle V, \Sigma, P, S\rangle$ mit $V = \{S, S_a, S_c, S_{ab}, S_{bc}\}$ und

$$
P = \left\{ \begin{array}{lllllll}
S & \rightarrow & S_{ab}S_c \mid S_aS_{bc} & & & & \\
S_a & \rightarrow & aS_a \mid a & & S_{ab} & \rightarrow & aS_{ab}b \mid ab \\
S_c & \rightarrow & cS_c \mid c & & S_{bc} & \rightarrow & bS_{bc}c \mid bc
\end{array} \right\}
$$

Ableitungen:

- $w = abc$:

  (i) $S \rightarrow S_{ab}\ S_c \rightarrow ab\ S_c \rightarrow ab\ c$

  (ii) $S \rightarrow S_a\ S_{bc} \rightarrow a\ S_{bc} \rightarrow a\ bc$

- $w = abbcc$: $S \rightarrow S_a\ {}_{,}S_{bc} \rightarrow a\ S_{bc} \rightarrow a\ bS_{bc}c \rightarrow a\ b\,bc\ c$

Aufgabe 3 & 4:
CNF-Design und CYK-Algorithmus

CHOMSKY-NORMALFORM

Eine kontextfreie Grammatik $G = \langle V, \Sigma, P, S \rangle$ ist in Chomsky-Normalform (CNF), wenn alle ihre Produktionsregeln eine der beiden folgenden Formen haben:

$$\mathsf{A} \to \mathsf{BC} \quad (\text{mit } \mathsf{B}, \mathsf{C} \in V) \qquad \text{oder} \qquad \mathsf{A} \to \mathtt{c} \quad (\text{mit } \mathtt{c} \in \Sigma)$$

Umwandlung in CNF:

(1) Eliminierung von ε-Regeln

(2) Eliminierung von Kettenregeln

(3) Extrahieren von Terminalsymbolen in Regeln $\mathsf{V}_\mathtt{c} \to \mathtt{c}$

(4) Reduzieren von Regeln der Form $\mathsf{A} \to \mathsf{B}_1 \cdots \mathsf{B}_\mathsf{n}$ auf $n = 2$

CYK: GRUNDIDEE

gegeben: kontextfreie Grammatik G in CNF
Frage: $w = \mathtt{a}_1 \cdots \mathtt{a}_n \in \mathbf{L}(G)$?

- Falls $|w| = 1$, dann ist $w \in \Sigma$ und es gilt:
  $w \in \mathbf{L}(G)$ genau dann wenn es eine Regel $S \to w$ in G gibt
- Falls $|w| > 1$, dann ist:
  $w \in \mathbf{L}(G)$ genau dann wenn es eine Regel $S \to AB$ und eine Zahl i gibt, so dass gilt

$$A \Rightarrow^* \mathtt{a}_1 \cdots \mathtt{a}_i \qquad \text{und} \qquad B \Rightarrow^* \mathtt{a}_{i+1} \cdots \mathtt{a}_n$$

Idee: Fall 2 reduziert das Problem $S \overset{?}{\Rightarrow}{}^* w$ auf zwei einfachere Probleme $A \overset{?}{\Rightarrow}{}^* \mathtt{a}_1 \cdots \mathtt{a}_i$ und $B \overset{?}{\Rightarrow}{}^* \mathtt{a}_{i+1} \cdots \mathtt{a}_n$, die man allerdings für alle Regeln $S \to AB$ und Indizes i lösen muss

CYK: PRAKTISCHE UMSETZUNG

Vorgehen: $V[i,j] =$ Menge aller A mit $A \Rightarrow^* w_{i,j}$

- Diagonale = Fall 1: existiert Terminalsymbolregel  $\to$ $\mathtt{a}$
- Fixiere Element : sei  in der gleichen Zeile ganz links und  direkt unten drunter
  - wenn eine Regel  $\to$ , dann füge  zu  hinzu
  - schiebe  nach rechts und  nach unten und wiederhole

Ist am Ende das Startsymbol $S \in V[1,|w|]$, dann $w \in \mathbf{L}$.

Beispiel: Wir betrachten das Wort $w = \mathtt{a} + \mathtt{b} \cdot \mathtt{c}$ der Länge $|w| = 5$.

$\mathtt{a}$	$V[1,1]$	$V[1,2]$	$V[1,3]$	$V[1,4]$	$V[1,5]$
$+$			$V[2,3]$	 $\cup$ 	$V[2,5]$
$\mathtt{b}$			$V[3,3]$		$V[3,5]$
$\cdot$				$V[4,4]$	$V[4,5]$
$\mathtt{c}$					$V[5,5]$
	$\mathtt{a}$	$+$	$\mathtt{b}$	$\cdot$	$\mathtt{c}$

AUFGABE 3 & 4

Aufgabe 3: Geben Sie für die nachfolgenden Sprachen L_i jeweils eine (kontextfreie) Grammatik G_i in CNF mit $L_i = L(G_i)$ an:

(a) Es sei L_1 genau die Menge der Palindrome über $\Sigma = \{a, b\}$.
(Palindrome sind Wörter, die vorwärts und rückwärts gelesen gleich sind, z.B. aba, $abba$, a, ε, bb)

(b) Es sei L_2 die Sprache aller $w \in \{a, b\}^*$ mit gleicher Anzahl an a's und b's.

(c) Es sei $L_3 = \{(ab)^n(ba)^n \mid n \geq 0\}$ über dem Alphabet $\Sigma = \{a, b\}$

Aufgabe 4: Gegeben seien L_1, L_2, L_3 wie in Aufgabe 3. Wenden Sie den CYK-Algorithmus für folgende Instanzen an um zu prüfen ob das jeweilige Wort zur entsprechenden Sprache gehört:

(a) L_1 mit $w = abba$ sowie $w = aba$

(b) L_2 mit $w = aababb$

(c) L_3 mit $w = ababbaba$