

THEORETISCHE GRUNDLAGEN DER INFORMATIK

TUTORIUM 7

WINTERSEMESTER 2014/15

MORITZ KLAMMLER

3. FEBRUAR 2015



Anmerkungen zum 6. Übungsblatt

- Macht „Unit-Tests“ für Eure Grammatiken.
- Schaut Euch noch einmal die Konstruktion einer regulären Grammatik *aus einem* endlichen Automaten an.
- Schaut Euch vielleicht noch einmal die „formale“ Methode aus der Vorlesung an, um eine kontextfreie Grammatik in CHOMSKY-Normalform zu überführen.

Tagesthemen

- Zusammenfassung Grammatiken & CHOMSKY-Hirarchie
- Nutzlose Variablen
- GREIBACH-Normalform

Zusammenfassung Sprachen, Maschinen, Grammatiken & CHOMSKY-Hirarchie

Sprachklasse	Grammatik	Maschine	Wortproblem	Abgeschlossenheit					
				$L_1 \cap L_2$	$L_1 \cup L_2$	$L_1 \cdot L_2$	L^*	L^R	L^C
regulär	Chomsky-Typ 3	NDFA	linear entscheidbar	✓	✓	✓	✓	✓	✓
kontextfrei	Chomsky-Typ 2	NDPDA	polynomiell entscheidbar	✗	✓	✓	✓	✓	✗
kontextsensitiv	Chomsky-Typ 1	LBNDTM	PSPACE-vollständig	✓	✓	✓	✓	✓	✓
entscheidbar	keine Entsprechung	Entscheider	entscheidbar	✓	✓	✓	✓	✓	✓
semi-entscheidbar	Chomsky-Typ 0	Akzeptor	semi-entscheidbar	✓	✓	✓	✓	✓	✗
allgemein	existiert nicht	existiert nicht	unentscheidbar	✓	✓	✓	✓	✓	✓

Nutzlose Variablen

Sei $G = (T, V, S, P)$ eine kontextfreie Grammatik. Eine Variable $A \in V$ heißt **nutzlos**, wenn es für kein $w \in T^*$ eine Ableitungsfolge

$$S \rightarrow^* \alpha A \beta \rightarrow^* w$$

gibt, in der A vorkommt.

Es gilt $L(G) = \emptyset$ genau dann wenn S nutzlos ist.

ROUTINE EliminateUselessVariables1

INPUT

$G = (T, V, S, P) ; ;$ kontextfreie Grammatik

OUTPUT

$V_1 ; ;$ sodass $V_1 = \{A \in V : \exists w \in T^* : A \rightarrow^* w\}$

P_1

BEGIN

$P_1 \leftarrow P$

$V_1 \leftarrow \{A \in V : \exists w \in T^* : (A, w) \in P_1\}$

$Q \leftarrow V_1$

WHILE $Q \neq \emptyset$ **DO**

FOREACH $A \in Q$ **DO**

$Q \leftarrow Q \setminus \{A\}$

FOREACH $(B, \alpha_1 A \alpha_2) \in P_1$ **DO**

$\tilde{P} \leftarrow \{(B, \alpha_1 w \alpha_2) : \exists w \in T^* : (A, w) \in P_1\}$

$P_1 \leftarrow P_1 \cup \tilde{P} \setminus \{(B, \alpha_1 A \alpha_2)\}$

IF $\tilde{P} \neq \emptyset$ **AND** $\alpha_1, \alpha_2 \in T^*$ **AND** $B \notin V_1$ **THEN**

$V_1 \leftarrow V_1 \cup \{B\}$

$Q \leftarrow Q \cup \{B\}$

FI

DONE

DONE

DONE

END

ROUTINE EliminateUselessVariables2**INPUT**

$G_1 = (T, V_1, S, P_1)$;; *kontextfreie Grammatik mit einigermaßen nützlichen Variablen V_1*

OUTPUT

V_2 ;; *sodass $V_2 = \{A \in V_1 : \exists \alpha_1, \alpha_2 \in (V_1 \cup T)^* : S \rightarrow^* \alpha_1 A \alpha_2\}$*

BEGIN

$V_2 \leftarrow \{S\}$

DO

$p_{\text{change}} \leftarrow 0$

FOREACH $A \in V_2$ **DO**

FOREACH $(A, \alpha_1 B \alpha_2) \in P_1$ **DO**

IF $B \notin V_2$ **THEN**

$V_2 \leftarrow V_2 \cup \{B\}$

$p_{\text{change}} \leftarrow p_{\text{change}} + 1$

FI

DONE

DONE

WHILE $p_{\text{change}} > 0$ **DONE**

END

ROUTINE EliminateUselessVariables

INPUT

$G = (T, V, S, P)$;; *kontextfreie Grammatik*

OUTPUT

$G' = (T, V', S, P')$;; *kontextfreie Grammatik ohne nutzlose Variablen außer S*

BEGIN

CALL EliminateUselessVariables1(G, V_1, P_1);

IF $S \notin V_1$ **THEN**

$V' \leftarrow \{S\}$

$P' \leftarrow \emptyset$

ELSE

$G_1 \leftarrow (T, V_1, S, P_1)$

CALL EliminateUselessVariables2(G_1, V_2);

$V' \leftarrow V_2$

$P' \leftarrow \{(A, \alpha) \in P : A \in V_2 \wedge \alpha \neq A\}$

FI

$G' \leftarrow (T, V', S, P')$

END

GREIBACH-Normalform

Sei $G = (T, V, S, P)$ eine kontextfreie Grammatik. G ist in **GREIBACH-Normalform (GNF)** genau dann wenn für alle $(\alpha, \beta) \in P$ gilt, dass

- $\alpha \in V$ und
- $\beta \in TV^*$.

Außerdem ist „manchmal“ der Spezialfall $\alpha = S$ und $\beta = \epsilon$ erlaubt, wenn S nie auf der rechten Seite einer Produktion steht.

Für jede kontextfreie Grammatik $G_1 = (T, V_1, S_1, P_1)$ mit $\epsilon \notin L(G_1)$ existiert eine kontextfreie Grammatik $G_2 = (T, V_2, S_2, P_2)$ in GREIBACH-Normalform, sodass $L(G_1) = L(G_2)$.

GREIBACH-Normalform

Um eine kontextfreie Grammatik ohne nutzlose Variablen in eine äquivalente Grammatik in GREIBACH-Normalform zu überführen, wechseln sich folgende Schritte ab, bis sich die Grammatik nicht mehr verändert:

0. Wenn auf der rechten Seite einer Produktion ein Terminal a vorkommt, das nicht das erste Zeichen ist, ersetze es durch ein neues Nichtterminal X_a und füge die Produktion $X_a \rightarrow a$ hinzu.
1. Streiche Produktionen $A \rightarrow B\gamma$, deren rechte Seite mit einem Nichtterminal $B \neq A$ beginnt, und nimm dafür $A \rightarrow \beta_1\gamma \mid \cdots \mid \beta_n\gamma$ für alle Produktionen $B \rightarrow \beta_1 \mid \cdots \mid \beta_n$ hinzu.
2. Streiche für jedes Nichtterminal A alle Produktionen $A \rightarrow A\gamma_1 \mid \cdots \mid A\gamma_n$ zugunsten von $A \rightarrow \alpha_1 Y_\gamma \mid \cdots \mid \alpha_m Y_\gamma$ für alle Produktionen $A \rightarrow \alpha_1 \mid \cdots \mid \alpha_m$ wo die α_j nicht mit A beginnen, und einem neuen Nichtterminal Y_γ . Nimm schließlich $Y_\gamma \rightarrow \gamma_1 \mid \cdots \mid \gamma_n$ und $Y_\gamma \rightarrow \gamma_1 Y_\gamma \mid \cdots \mid \gamma_n Y_\gamma$ hinzu.