

Übung 3

Grammatiken - Formale Sprachen

Einführung in Grammatiken

Chomsky-Grammatiken wurden 1959 von dem amerikanischen Wissenschaftler Noam Chomsky eingeführt. Sie sind z.B. in einer bestimmten Form bei der Definition von Programmiersprachen von großer Bedeutung. Eine (Chomsky-)Grammatik G definiert man formal als Viertupel

$$G = (N, T, P, S)$$

für das gilt:

- N ist eine endliche, nichtleere Menge von Zeichen, den sog. *Nichtterminalsymbolen* (werden durch Terminalsymbole ersetzt).
- T ist eine endliche, nichtleere Menge von Zeichen, den sog. *Terminalsymbolen*. N und T sind *disjunkt*.
- S ist ein Element von N und heißt *Startsymbol* oder *Axiom*.
- P ist eine endliche Menge von Paaren der Form (p, q) , wobei gilt: p und q sind Wörter, die jeweils aus der Aneinanderreihung endlich vieler, nicht notwendigerweise verschiedener Terminal- und Nichtterminalsymbolen entstanden sind. p muss *mindestens ein* Nichtterminalsymbol enthalten. P nennt man Produktionssystem oder Regelsystem, die Elemente von P werden *Produktionen*, Produktionsregeln, Ersetzungsregeln oder schlicht *Regeln* genannt.

Ein einfaches Beispiel

An folgendem einfachen Beispiel soll die Verwendung von Grammatiken kurz erläutert werden:

Gegeben seien die Grammatik $G = (N, T, P, S)$ mit

$N = \{S, A, B\}, T = \{g, d, i\}$ und

$P = \{(S, gA), (A, dB), (B, iS), (B, i)\}$

Bestimmen Sie (mit Hilfe von Beispielen) die Sprache $L(G)$.

Der einfachste Weg zu bestimmen, wie die Sprache $L(G)$ lautet, ist es, einige Wörter mit den Produktionen P zu bestimmen. Begonnen wird immer mit dem Startsymbol S . Wir suchen also nach einer Produktionsvorschrift bei der das S auf der linken Seite steht.

Regel 1: (S, gA) . Das Startsymbol S wird durch gA ersetzt. $S \rightarrow gA$. Unser aktuelles Wort lautet nun also gA . Im Wort gA befindet sich das Nichtterminalsymbol A . Wir suchen nun also nach einer Regel bei der ein A auf der linken Seite der Produktionsvorschrift steht $\rightarrow$ Regel 2: (A, dB) .

Auf diese Weise wendet man weiter die verschiedenen Produktionsvorschriften an bis **kein** Nichtterminal mehr im gebildeten Wort vorhanden ist.

$$S \rightarrow gA \rightarrow gdB \rightarrow gdiS \rightarrow gdi gA \rightarrow gdi gdB \rightarrow gdi gdi$$

Nach Anwenden einiger Regeln sollte an den produzierten Worten die Sprache L erkennbar sein. In diesem Fall lässt sich mit $L(G)$ das Wort **gdi** mindestens einmal, beliebig oft hintereinander darstellen!

Achtung, geändert: Ursprünglich war in Aufgabe 1 und 2 ein Terminal c angegeben, das eigentlich das Leere Wort ε hätte sein sollen.

Aufgabe 1: Sprachen und Grammatiken I (3 Punkte)

Hinweis: ε ist ein Nichtterminal und bedeutet „Leeres Wort“ (http://de.wikipedia.org/wiki/Leeres_Wort), also „nichts“.

Gegeben seien die Grammatiken $G_i = (N, T, P_i, S)$ mit

$N = \{S, B\}, T = \{a, b, \varepsilon\}$ und

$P_2 = \{(S, aB), (S, bB), (B, aS), (B, bS), (B, \varepsilon)\}$

$P_3 = \{(S, aSa), (S, bSb), (S, \varepsilon)\}$

Bestimmen Sie (mit Hilfe von Beispielen) die Sprache $L(G_2)$ und $L(G_3)$.

Aufgabe 2: Sprachen und Grammatiken II (3 Punkte)

Sei $G_i = (N, T, P_i, S)$ gegeben mit $N = \{S, B\}, T = \{a, b, \varepsilon\}$

Erläuterung der mathematischen Schreibweise:

$L(G_4) = \{w | w = a^n b^n, n \in \mathbb{N}_0\}$

Beispiele hierfür sind: aaaabbbb ; aabb $\rightarrow$ n-mal a gefolgt von n-mal b

$L(G_5) = \{w | w = (ab)^n, n \in \mathbb{N}\}$

Beispiele hierfür sind: ababab ; ababababab $\rightarrow$ n-mal ab hintereinander

a) Bestimmen Sie P_4 (möglichst minimal) so, dass gilt: $L(G_4) = \{w | w = a^n b^n, n \in \mathbb{N}_0\}$

b) Bestimmen Sie P_5 (möglichst minimal) so, dass gilt: $L(G_5) = \{w | w = (ab)^n, n \in \mathbb{N}\}$

Aufgabe 3: EBNF I (2 Punkte)

Ermitteln Sie, ob folgende Worte mit den gegebenen EBNFs erzeugt werden können:

	$\{a\} \{b\}$	$\{a\}b[a]$	$[a]\{(ab)\}[b]\{a\}$
aba			
aab			
aaab			
ba			
ε			

Aufgabe 4: EBNF II (2 Punkte)

Für die Syntax von E-Mail-Adressen (adress) liegt folgender Vorschlag in EBNF vor.

address = $\langle name \rangle . \langle name \rangle [. \langle name \rangle] @ \langle domain \rangle$

name = $\langle letter \rangle \{ \langle symbol \rangle \}$

letter = $a | b | \dots | z | A | B | \dots | Z$

symbol = $\langle letter \rangle | \langle digit \rangle$

digit = $0 | 1 | \dots | 9$

domain = $\langle dotnum \rangle | ([\langle subdomain \rangle .] \langle topleveldomain \rangle)$

dotnum = $"[" \langle snum \rangle . \langle snum \rangle . \langle snum \rangle . \langle snum \rangle "]"$

snum = $0 | 1 | \dots | 255$

subdomain = $\langle name \rangle [- \langle name \rangle]$

topleveldomain = $com | edu | gov | org | \langle countrycodetld \rangle$

countrycodetld = $aq | de | es | sy$

Sind die folgenden Beispiele im Sinne der angegebenen Syntax zulässige E-Mail-Adressen?
Geben Sie **alle** Fehler an, falls eine Adresse nicht zulässig ist.

- Mina.Ralwasser@gmx.gov
- Klaus.Tro.Phobie@ncc-1701.org
- ali.mente@t-online.business.net

- trompeter@informatik.uni-kl.de
 - 2742.open22@ich-bin-xena.com
-

Hinweise zur Abgabe:

- Abgabe der Lösungen für diese Übung per E-Mail an Herrn Marc Beck
<marc.beck@hs-kl.de>
bis **Dienstag, 21.04.2015, vor Beginn Ihrer Übung.**
- Bitte geben Sie als „Betreff“ an: **Abgabe GDI Übung 3 SS2015 *Ihr Name*.**