Formale Systeme, Automaten, Prozesse, SS 2020
Prof. Dr. P. Rossmanith
J. Dreier, T. Hartmann, H. Lotze, D. Mock

Datum: 25.05.2020

Übungsblatt 07

Aufgabe T21
Sei $G = (N, T, P, S)$ folgende kontextfreie Grammatik mit $N = \{S, A, B, C, D\}$, $T = \{a, b, c, d\}$ und P :

$$\begin{array}{lcl} S & \to & BDA \mid DAC \mid DC \\ A & \to & SA \mid aA \\ B & \to & \epsilon \mid Sc \mid cD \mid AD \\ C & \to & aD \mid dA \mid bC \mid a \mid d \mid b \\ D & \to & BS \mid cC \mid aA. \end{array}$$

Dies ist die gleiche Grammatik wie die von letzter Woche, nur dass es für A weniger Produktionen gibt.

a) Ist $L(G)$ endlich?

b) Ist $L(G)$ universell, d.h. $L(G) = \{a, b, c, d\}^*$?

Aufgabe T22
Bringen Sie die folgende Grammatik $G = (N, T, P, S)$ erst in Chomsky- und dann in Greibach-Normalform. Hier ist $N = \{S, A, B, C\}$, $T = \{a, b, c\}$ und P:

$$S \to ABC \mid a, \quad A \to Sa, \quad B \to Sb, \quad C \to cS.$$

Aufgabe T23
Es sei $L = \{\, p\$p\$p \mid p \in \{a, b\}^* \,\}$. Benutzen Sie das Pumping-Lemma um zu zeigen, dass L nicht durch eine kontextfreie Grammatik erzeugt werden kann.

Aufgabe H19 (10 Punkte)
Schreiben Sie ein Programm, welches boolsche Ausdrücke evaluieren kann, wie sie in Aufgabe H18 definiert wurden. Es bietet sich an einen *recursive descent parser* zu entwickeln, ähnlich zu jenem der in der Vorlesung für arithmetische Ausdrücke vorgestellt wurde. Zu einem booleschen Ausdruck sollte die Ausgabe `True` oder `False` erzeugt werden, wenn der Ausdruck syntaktisch korrekt ist und andernfalls ein Fehler gemeldet werden.
Führen Sie Ihr Programm für alle Ausdrücke aus, die in den Zeilen der Datei `boolsche-ausdruecke` enthalten sind, welche sich im Moodleraum befindet. Dort finden sich Zeilen wie zum Beispiel `~1|(1|0&(0))&1`, welche dem boolschen Ausdruck $\neg 1 \vee (1 \vee 0 \wedge (0)) \wedge 1$ entspricht. Geben Sie die Ausgabe des Programms mit ab.

Hinweis: Verwenden Sie eine Grammatik, die nicht links-rekursiv ist.

Bonuspunkte: Überlegen Sie sich, wie möglichst nützliche Fehlermeldungen erzeugt werden können. Diese sollten die Stelle und die Art des Fehlers anzeigen. Erläutern Sie Ihre Entscheidungen und deren Implementierung. Zeigen Sie viele aussagekräftige Beispiele. Die auf der Webseite vorgegebenen Ausdrücke sind alle syntaktisch korrekt.

Aufgabe H20 (10 Punkte)
In der Aufgabe H18 entwarfen wir kontextfreie Grammatiken für boolsche Ausdrücke. Erzeugen Sie nun kontextfreie Grammatiken in Chomsky- und Greibach-Normalform für jene Sprache.

Aufgabe H21 (10 Punkte)
Betrachten Sie folgende Aufgabe.

Aufgabe I1

Beweisen oder widerlegen sie die folgenden Aussagen.

a) Die Sprache $L_1 = \{\, a^i b^j \mid i + j \leq 200, i + j \geq 100 \,\}$ ist nicht regulär.
b) Die Sprache $L_2 = \{(abc)^n \mid n \in \mathbf{N}\}$ ist regulär.
c) Die Sprache $L_3 = \{a^n b^n a^n b^n \mid n \geq 0\}$, mit $\Sigma = \{a, b\}$ ist kontextfrei.

Die drei Studierenden Henri Lotse, Kim Hartfrau und David Rock geben zusammen ihre Hausaufgaben in FoSAP ab, und haben eine Lösung zu Aufgabe I1 eingereicht.
Sie sind nun ein FoSAP-Tutor und müssen die Abgabe der drei korrigieren. Wie viele Punkte würden Sie vergeben? Seien Sie dabei genauso streng wie Ihre eigene Tutorin.

a)

L_1 *nicht regulär: Sei* $15 \leq n \leq 100$ *Sei* $w = a^n b^n$. $w \in L$. *Wir betrachten eine Zerlegung* $w = xyz$ *mit* $|xy| \leq n$ *und* $|y| > 0$. *Wegen Pumping–Lemma muss auch* $xy^i z \in L$ *für alle* i. *Das Wort* $xy^{200}z$ *hat Länge ungefähr* 200 *und somit zu viele* a *und* b. *So gilt gilt* $xy^{200}z \notin L$. *Also* L *nicht regulär und Aussage wahr.*

b)

Sei n *eine große und schöne Zahl und* $w = (abc)^n$ *ein Wort. Mit Leichtigkeit kann man herausfinden, dass dieses Wort in der Sprache enthalten ist und länger als* n *ist. Wir wählen die Zerlegung so dass* x *und* z *das leere Wort sind und* y w *ist. Damit gelten Bedingungen (1) und (2) des Pumpinglemmas. Betrachten wir das gepumpte Wort* $xy^i z$ *für jedes natürliche* i. *Dies gleicht* $((abc)^n)^i = (abc)^{in}$. *Damit ist es in der Sprache enthalten und das Pumpinglemma gilt für* L. *Also muss* L *regulär sein.*

c)

$z = a^n b^n a^n b^n$, *dann gibt's Zerlegung* z *mit* $uvwxy$ *mit* $|vwx| \leq n$ *und* $|vx| > 0$. *Sei jetzt* $uvwxy$ *so das der* vwx*-Part im ersten viertel liegt, also nur die ersten* a*s enthält. Nach dem Pump-Lemma muss dann auch* $uv^i wx^i y \in L$ *für alle* i *kann aber trivialistischlischterweise nicht sein da die Anzahl der* a *am Anfang sich ändert und am Ende des Wortes aber gleich blieb.* $\Rightarrow$ L_3 *nicht konteckstfrei und die Ausage widerlegt.*