Institut für Informatik
M. Mutz

Übungsblatt 6

Theoretische Informatik

Bearbeitungszeit: 20.05.2024 bis 26.05.2024, 16:00 Uhr
Besprechung: 27.05.2024, 10:30 Uhr in Hörsaal 5E

Abgabe: als PDF über das ILIAS
Gruppenabgaben möglich und erwünscht

Aufgabe 1 (CYK-Algorithmus I) 10 Punkte
Gegeben sei die Grammatik $G' = (\Sigma, N', S, P')$ mit $N' = \{S, A, B, C, D_1, D_2, X_0, X_1\}$ und

$$\begin{aligned} P' = \{S &\to X_0C \mid X_0X_0, \\ C &\to X_0A, \\ A &\to X_1D_1 \mid X_1D_2 \mid X_0B \mid 0, \\ D_1 &\to AD_2, \\ D_2 &\to X_1X_1, \\ B &\to X_0B \mid 0, \\ X_0 &\to 0, \\ X_1 &\to 1\}. \end{aligned}$$

a) Entscheiden Sie mit dem Algorithmus von Cocke, Younger und Kasami (CYK-Algorithmus), ob die Wörter

   i) $w_1 = 0011$ und
   ii) $w_2 = 001011$

   in $L(G')$ liegen. Geben Sie die Tabelle bzw. Dreiecksmatrix dabei jeweils vollständig an. *Bitte beachten Sie, dass Ihre Grammatik in CNF sein muss, damit Sie den CYK-Algorithmus anwenden können.*

b) Geben Sie $L(G')$ formal als Menge von Wörtern an. Die Angabe soll als in sich abgeschlossener Mengenausdruck geschehen. Referenzieren Sie keine weiteren Konstrukte wie die Grammatik G', Teile von G', wie die Produktionsregeln P', einen regulären Ausdruck, einen Automaten, oder eine andere Grammatik. Die Ausnahme hier stellt Σ dar, welches Sie referenzieren dürfen. Lösungen wie $L(G') = \{w | w \in L(...)\}$ führen zu 0 Punkten.

Aufgabe 2 (CYK-Algorithmus II) 10 Punkte
Gegeben seien das Alphabet $\Sigma = \{a, b, c\}$ und die folgende Tabelle, welche mit dem CYK-Algorithmus für das Wort $w = aabbcc$ erstellt wurde. Aus dem Ergebnis kann gefolgert werden, dass w in der Sprache liegt, welche die zugrundeliegende Grammatik erzeugt.

i	1	2	3	4	5	6
5	S, Z					
4	Y	Y				
3	X		Z, S			
2	A		Y	Y		
1		X		Z	Z, S	
0	A	A	B	B	Y	Y
j	a	a	b	b	c	c

Geben Sie eine mögliche Grammatik G an, welche diese Tabelle erzeugen kann.

Aufgabe 3 (Kellerautomat I)10P
Der Kellerautomat $M = (\{a, b\}, \{\#, A, B\}, \{z_0, z_1, z_2, z_3, z_4\}, \delta, z_0, \#)$ verfügt über die folgenden Regeln um δ zu definieren:

$$\begin{array}{rll} z_0a\# \rightarrow z_1AA\# & z_1bA \rightarrow z_1\lambda & z_2\lambda\# \rightarrow z_0\# \\ z_0b\# \rightarrow z_2B\# & z_1\lambda\# \rightarrow z_0\# & z_3\lambda B \rightarrow z_2\lambda \\ z_0\lambda\# \rightarrow z_4\lambda & z_2aB \rightarrow z_3\lambda & z_3\lambda\# \rightarrow z_1A\# \\ z_1aA \rightarrow z_1AAA & z_2bB \rightarrow z_2BB & \end{array}$$

a) Liegt das Wort $bababb$ in der Sprache des Kellerautomaten $L(M)$? Verwenden Sie in Ihrer Argumentation die entsprechende(n) Konfigurationsfolge(n).

b) Geben Sie jeweils die Inhalte des Kellers an, nachdem M die folgenden Zeichenketten gelesen hat:

   i) $b^3ab^4a^3$

   ii) a^2b^3

c) Beschreiben Sie die Sprache $L(M)$ möglichst formal. Die Angabe soll als in sich abgeschlossener Mengenausdruck geschehen. Referenzieren Sie keine weiteren Konstrukte wie den Kellerautomaten M, Teile von M, wie die Überführungsfunktion δ, die Relation von Konfigurationenfolgen, einen regulären Ausdruck, eine Grammatik, oder einen anderen Automaten. Die Ausnahme hier stellt $\Sigma = \{a, b\}$ dar, welches Sie referenzieren dürfen. Lösungen wie $L(M) = \{w | w \in L(...)\}$ führen zu 0 Punkten.

d) Zeigen Sie, dass die Überführungsfunktion δ von M nicht deterministisch ist.

Aufgabe 4 (Abschlusseigenschaften)10P

(a) Betrachten Sie eine Sprache L und einen DFA $M = (\Sigma, Z, \delta, z_0, F)$ mit totaler Überführungsfunktion δ und mit $L = L(M)$.
Geben Sie einen DFA M' mit $\overline{L} = L(M')$ als Tupel an, um zu zeigen, dass die Klasse der regulären Sprachen unter Komplement abgeschlossen ist.
Hinweis: Ihr DFA M' soll nicht konkret angegeben werden, sondern muss aus jedem beliebigen Automaten M, wie er in der Aufgabenstellung angegeben ist, konstruierbar sein.

(b) Zeigen Sie nur mit Hilfe von Abschlusseigenschaften für kontextfreie Sprachen und Satz 3.36 folgende Behauptungen. Sie dürfen Sprachen, die von regulären Ausdrücken erzeugt werden, direkt als regulär verwenden. Aus der Vorlesung bekannte Sprachen dürfen wieder direkt verwendet werden (z.B. bewiesene Behauptungen aus dem Skript zu spezifischen Sprachen sind also auch erlaubt zu nutzen).

$$L_1 = \{a^i b^{i+j} a^j \mid i, j \geq 0\} \in CF \text{ mit } L_1 \subseteq \{a, b\}^*$$
$$L_2 = \{a^n b^m c^{max(n,m)} | n, m \geq 0\} \notin REG \text{ mit } L_2 \subseteq \{a, b, c\}^*$$