Institut für Theoretische Informatik
Lehrstuhl Prof. Dr. D. Wagner

Übungsblatt 7

Vorlesung Theoretische Grundlagen der Informatik im WS 20/21

Ausgabe: 5. Februar 2021
Abgabe: 16. Februar 2021, 11:30 Uhr (digital im ILIAS)

Aufgabe 1

(3 + 1 + 1 = 5 Punkte)

Sei $G = (\Sigma, V, S, R)$ mit $\Sigma = \{\mathsf{a}, \mathsf{b}\}$ und $V = \{S, A, B, C, D, E\}$ die durch folgende Regelmenge gegebene Grammatik:

$$\begin{aligned}
S &\rightarrow \mathsf{b}E \mid EA \\
A &\rightarrow CB \mid A\mathsf{a} \\
B &\rightarrow S \mid \mathsf{b} \\
C &\rightarrow CA \mid AD \\
D &\rightarrow D\mathsf{a} \mid \varepsilon \\
E &\rightarrow \mathsf{b}B \mid \varepsilon
\end{aligned}$$

(a) Identifizieren Sie alle nutzlosen Variablen in G mit dem Verfahren aus der Vorlesung. Geben Sie die Grammatik G' an, die durch Entfernen der nutzlosen Variablen entsteht.

(b) Welche Sprache erzeugt G? Welchen Chomsky-Typ hat $L(G)$? Ist $L(G)$ endlich?

(c) In der Vorlesung wurde ein Verfahren vorgestellt, um zu entscheiden, ob eine gegebene kontextfreie Grammatik endlich ist. Dabei werden zunächst alle nutzlosen Variablen entfernt. Warum ist das nötig? Geben Sie ein Beispiel an, wo das Verfahren ein falsches Ergebnis liefert, wenn die nutzlosen Variablen vorher nicht entfernt werden.

Aufgabe 2

(2 + 2 + 1 = 5 Punkte)

In dieser Aufgabe betrachten wir Kellerautomaten mit Reset-Zustand (RPDA). Diese arbeiten wie normale nichtdeterministische Kellerautomaten (NPDA), verfügen aber zusätzlich über einen *Reset-Zustand* $r \in Q$. Wenn ein RPDA in den Zustand r übergeht, wird der Stackinhalt zurückgesetzt (d.h. der Stack wird geleert und das Initialsymbol Z_0 auf den Stack gelegt) und der Lesekopf springt zurück zum ersten Zeichen der Eingabe. Der RPDA bleibt dabei im Zustand r. Sie können in dieser Aufgabe davon ausgehen, dass sowohl bei NPDAs als auch bei RPDAs das Ende der Eingabe mit einem speziellen Symbol markiert ist.

(a) Beschreiben Sie, wie die Sprache $L = \{\mathsf{a}^n\mathsf{b}^n\mathsf{c}^n \mid n \in \mathbb{N}\}$ von einem RPDA erkannt werden kann.

(b) Seien zwei beliebige kontextfreie Sprachen L_1 und L_2 gegeben. Beschreiben Sie, wie die Sprache $L_1 \cap L_2$ von einem RPDA erkannt werden kann.

(c) Wie kann das Maschinenmodell erweitert werden, sodass der Schnitt von beliebig (aber endlich) vielen kontextfreien Sprachen erkannt werden kann?

Die Grundstruktur eines Kellerautomaten soll dabei erhalten bleiben. Lösungen der Art „Baue den Kellerautomaten zu einer Turing-Maschine um" werden nicht akzeptiert.

Aufgabe 3 (3 + 1 = 4 Punkte)

Sei L eine kontextfreie Sprache. Die Funktionen α und β sind wie folgt definiert.

$$\alpha(L) = \{yx \mid xy \in L\} \tag{1}$$
$$\beta(L) = \{yxz \mid xyz \in L\} \tag{2}$$

Zeigen Sie:

- Die Menge der kontextfreien Sprachen ist abgeschlossen unter α.
- Die Menge der kontextfreien Sprachen ist nicht abgeschlossen unter B.

Aufgabe 4 (3 Punkte)

Sei L_1 eine kontextfreie Sprache und sei L_2 eine reguläre Sprache. Zeigen Sie, dass $L_1 \cap L_2$ eine kontextfreie Sprache ist.

Aufgabe 5 (1 + 2 + 1 = 4 Punkte)

Wir betrachten die Buchstaben $7 \times B, 5 \times A, 4 \times L, 2 \times E, 1 \times T$.

(a) Geben Sie einen Huffman-Code für die gegebene Verteilung an. Verwenden Sie die `0`, wenn Sie zu einem häufigeren Buchstaben absteigen und die `1`, wenn Sie zu einem weniger häufigen Buchstaben absteigen.

(b) Wir ziehen mit Zurücklegen fünf Buchstaben und kodieren das so entstandene Wort mit der Huffman-Codierung. Wie lang ist die Codierung erwartet?

(c) Decodieren Sie das Wort `111110110`. Ist das Ergebnis eindeutig?

Aufgabe 6 (1 + 6 = 7 Punkte)

Für eine Menge U von Variablen bezeichnen wir eine Klauselmenge $\mathcal{C}$ als

- 3-SAT-*Klauselmenge*, wenn jeden Klausel höchstens drei Literale enthält,
- UNIQUE-3-SAT-*Klauselmenge*, wenn zusätzlich jede Variable höchstens ein Mal in $\mathcal{C}$ vorkommt (insbesondere kommt nur eines von u und $\bar{u}$ vor und keine Klausel enthält eine Variable doppelt).
- 2-SAT-*Klauselmenge*, wenn jede Klausel höchstens zwei Literale enthält.

Daraus ergeben sich folgende SAT-Varianten:

Gegeben: Menge U von Variablen und Klauselmenge über U wie folgt:

- für 2-SAT: eine 2-SAT-Klauselmenge
- für 3-SAT: eine 3-SAT-Klauselmenge
- für UNIQUE-3-SAT: eine UNIQUE-3-SAT-Klauselmenge
- für UNIQUE-3-2-SAT: eine UNIQUE-3-SAT-Klauselmenge und eine 2-SAT-Klauselmenge

Frage: Existiert eine erfüllende Wahrheitsbelegung?

Von 3-SAT wissen Sie bereits, dass das Problem $\mathcal{NP}$-vollständig ist. Ebenso wissen Sie, wie man 2-SAT in Polynomialzeit lösen kann.

(a) Zeigen Sie, dass UNIQUE-3-SAT in $\mathcal{P}$ liegt.

(b) Zeigen Sie, dass UNIQUE-3-2-SAT $\mathcal{NP}$-vollständig ist. Geben Sie dabei explizit an, welches Problem Sie auf welches reduzieren. Verwenden Sie für die Reduktion das Problem 3-SAT.