

Einführung in die Theoretische Informatik  
Sommersemester 2020 – Übungsblatt 5

**AUFGABE 5.1.** (*Wichtige Begriffe*)

Stufe A

Überprüfen Sie, dass Sie die folgenden Begriffe korrekt definieren können.

- nützlich, erzeugend, erreichbare Symbole
- Chomsky-Normalform
- CYK-Algorithmus
- Pumping-Lemma für kontextfreie Sprachen
- Präfix, Suffix, Infix

**AUFGABE 5.2.** (*Chomsky-Normalform*)

Stufe C

**Chomsky-Normalform Onlinetool**

Sie können das Überführen von Grammatiken in Chomsky-Normalform auf der Website <http://grammar.epfl.ch/> üben. Beachten Sie dabei, dass die Website Feedback gibt, ob Ihre Lösung richtig oder falsch ist, aber nicht überprüft ob Sie das Verfahren aus der Vorlesung verwendet haben.

Die CFG  $G$  bestehe aus folgenden Produktionen über dem Alphabet  $\Sigma = \{a, b\}$ :

$$\begin{aligned} S &\rightarrow ASA \mid aB \\ A &\rightarrow B \mid S \mid CB \\ B &\rightarrow b \mid \varepsilon \\ C &\rightarrow aC \\ D &\rightarrow aSCb \mid a \end{aligned}$$

- Beschreiben Sie in eigenen Worten, wann ein Nichtterminal *nützlich* in einer Grammatik ist.
- Reduzieren Sie die Grammatik  $G$  auf die nützlichen Nichtterminale.
- Überführen Sie die reduzierte Grammatik dann in Chomsky-Normalform.
- Erklären Sie in eigenen Worten, wie Sie nach Überführen einer Grammatik in CNF überprüfen können, dass Sie keine Fehler gemacht haben.<sup>1</sup>

**AUFGABE 5.3.** (*CYK-Algorithmus*)

Stufe C

Die folgende Aufgaben können Sie auch im AutomataTutor unter der Kategorie “CYK Algorithm” finden. Beachten Sie, dass sich dort auch eine Hausaufgabe mit limitierten Versuchen befindet.

Wir betrachten die Grammatik  $G = (\{S, T, U, A, B, C\}, \{a, b, c\}, P, S)$  in CNF mit den folgenden Produktionen  $P$ :

$$\begin{aligned} S &\rightarrow TS \mid CT \mid a & A &\rightarrow a \\ T &\rightarrow AU \mid TT \mid c & B &\rightarrow b \\ U &\rightarrow SB \mid AB & C &\rightarrow c \end{aligned}$$

Bestimmen Sie mit dem CYK-Algorithmus, ob  $ccaab \in L(G)$  und  $aabcc \in L(G)$ . Geben Sie dabei auch die berechneten Tabellen an.

**AUFGABE 5.4.** (*Pumping-Lemma für Kontextfreie Sprachen*)

Stufe C

Wir betrachten Pfeil-Sprachen über dem Alphabet  $\Sigma = \{\uparrow, \downarrow, \leftarrow, \rightarrow\}$ . Wir interpretieren dabei ein Wort  $w \in \Sigma^*$  als einen Pfad in einem 2D-Gitter.

- Geben Sie eine formale Definition für die folgenden beiden Sprachen an:
  - Pfade, die in den Ursprung zurückkehren.
  - Pfade, die „in großer Kurve umkehren“ — *beliebig weit* nach rechts fahren, dann *noch weiter* entweder nach oben oder unten gehen und letztlich wieder umkehren und *noch weiter* nach links fahren. Diese Pfade enden dann links vom Ursprung.
- Zeigen Sie mit Hilfe des Pumping-Lemma für kontextfreie Sprachen, dass diese Sprachen nicht kontextfrei sind

**AUFGABE 5.5.**

Stufe D

<sup>1</sup>Hier geht es nicht um einen formalen Beweis, dass die beiden Sprachen gleich sind, sondern um eine Strategie, wie Sie bei Aufgaben wie im vorherigen Aufgabenteil Fehler vermeiden.

---

Sei  $\Sigma$  ein Alphabet und  $L \subseteq \Sigma^*$ .  $L$  ist *präfix-abgeschlossen*, wenn für alle  $w \in L$  und  $v \in \Sigma^*$  mit  $v \preceq w$  (also  $v$  ist ein Präfix von  $w$ ) gilt:  $v \in L$ .

*Beweisen Sie, dass jede unendliche, präfix-abgeschlossene, kontextfreie Sprache eine unendliche, reguläre Teilmenge enthält.*

**Bemerkung:** Zu dieser Aufgabe gibt es keine Videolösung. Eine textuelle Lösung finden Sie vorab auf Moodle/der Vorlesungswebsite.

**AUFGABE 5.6.** (*Schnitt regelt*)

Beweisen Sie, dass kontextfreie Sprachen unter Schnitt mit regulären Sprachen abgeschlossen sind.

Stufe D