

Kennzahl: \_\_\_\_\_

Kennwort: \_\_\_\_\_

Arbeitsplatz-Nr.: \_\_\_\_\_

---

**Herbst  
2007**

**46113**

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen**  
**— Prüfungsaufgaben —**

---

Fach: **Informatik (Unterrichtsfach)**

Einzelprüfung: **Theoretische Informatik**

Anzahl der gestellten Themen (Aufgaben): 2

Anzahl der Druckseiten dieser Vorlage: 5

---

Thema Nr. 1

Es sind alle Teilaufgaben zu bearbeiten!

Aufgabe 1 (endliche Automaten)

Gegeben sei ein nichtdeterministischer endlicher Automat  $A_1$  mit Zustandsmenge  $Q = \{A, B, C, D, E\}$ , Eingabealphabet  $\Sigma = \{0, 1\}$ , Startzustand  $A$  und Endzustandsmenge  $F = \{E\}$ .

Die Übergangsfunktion  $\delta : Q \times \Sigma \cup \{\epsilon\} \rightarrow P(Q)$  sei durch folgende Tabelle gegeben:

$\delta$	$A$	$B$	$C$	$D$	$E$
0	$\{C\}$	$\emptyset$	$\{A\}$	$\{E\}$	$\{D\}$
1	$\{C\}$	$\{B\}$	$\{D\}$	$\{A\}$	$\{E\}$
$\epsilon$	$\{B\}$	$\emptyset$	$\emptyset$	$\{B\}$	$\emptyset$

Dabei bezeichne  $P(Q)$  die Potenzmenge von  $Q$ .

- a) Zeichnen Sie den Automaten in Form eines Übergangsdiagramms, das für jeden Zustand einen Knoten und Kanten zwischen den Knoten gemäß der Übergangsfunktion enthält. Markieren Sie den Endzustand gesondert.
- b) Konstruieren Sie mit Hilfe der Teilmengenkonstruktion einen zu  $A_1$  äquivalenten deterministischen endlichen Automaten  $A_2$ . Erzeugen Sie dazu vom Startzustand von  $A_1$  ausgehend schrittweise die Zustände des deterministischen endlichen Automaten  $A_2$  als Teilmengen der Zustände von  $A_1$ . Geben Sie die jeweils erzeugten Teilmengen an. Zeichnen Sie den so erhaltenen Automaten  $A_2$  als Übergangsdiagramm.
- c) Identifizieren Sie äquivalente Zustände von  $A_2$ , indem Sie den Table-Filling-Algorithmus anwenden. Bauen Sie dazu schrittweise eine Tabelle auf, in der für die möglichen Zustandspaare vermerkt ist, ob sie unterscheidbar sind. Begründen Sie die Schritte des Algorithmus.
- d) Konstruieren Sie einen zu  $A_2$  äquivalenten deterministischen endlichen Automaten, der eine minimale Anzahl von Zuständen enthält. Fassen Sie dazu alle äquivalenten Zustandspaare von  $A_2$  zu jeweils einem Zustand zusammen. Zeichnen Sie den resultierenden Automaten als Übergangsdiagramm.

**Aufgabe 2:****(Wortproblem für kontextfreie Grammatiken)**

Gegeben sei folgende kontextfreie Grammatik in Chomsky-Normalform:

$$G = (\{A, B, C, S\}, \{a, b\}, P, S)$$

mit Variablenmenge  $V = \{A, B, C, S\}$ , Alphabet  $\Sigma = \{a, b\}$  und Startsymbol  $S$ . Die Produktionen  $P$  sind gegeben durch:

$$P = \{ \begin{array}{l} S \rightarrow AA \mid BC \\ A \rightarrow BA \mid a \\ B \rightarrow CB \mid b \\ C \rightarrow AB \mid a \end{array} \}$$

- Beweisen Sie, dass diese Grammatik mehrdeutig ist, d. h. dass es für mindestens ein Wort der von  $G$  erzeugten Sprache mindestens zwei unterschiedliche Ableitungsbäume gibt.
- Entscheiden Sie durch Anwendung des CYK-Algorithmus, ob das Wort  $w = baab$  von der Grammatik  $G$  erzeugt wird. Konstruieren Sie dazu schrittweise eine Tabelle, deren Einträge  $X_{ij}$  alle Variablen der Grammatik enthält, die das Teilwort  $a_i \dots a_j$  des gegebenen Wortes  $w = a_1 \dots a_n$  erzeugen können. Geben Sie die zur Ermittlung jedes Eintrages erforderlichen Berechnungen an und begründen Sie die Berechnungen.

**Aufgabe 3:****(Berechenbarkeit und Turingmaschinen)**

- Konstruieren Sie eine deterministische Turingmaschine, die eine Eingabe  $x \in \{0,1\}^*$  als Binärzahl interpretiert und zu  $x$  die Zahl 1 addiert. Beim Start der Turingmaschine stehe das Eingabewort  $x$  auf dem Eingabeband. Beschreiben Sie das Vorgehen und die Arbeitsweise der von Ihnen entworfenen Turingmaschine. Erläutern Sie die Rolle der Zustände der von Ihnen entworfenen Turingmaschine und geben Sie die Übergangsfunktion in Tabellenform an. Zeichnen Sie die Turingmaschine als Übergangsdiagramm.
- Illustrieren Sie die Arbeitsweise der von Ihnen konstruierten Turingmaschine, indem Sie die Berechnungsschritte der Turingmaschine für die Eingabe 101 angeben. Verwenden Sie dazu die Beschreibung durch Konfigurationsübergänge.
- Turingmaschinen können dadurch erweitert werden, dass das Eingabeband in mehrere Spuren unterteilt wird. Geben Sie an, ob durch diese Erweiterung die Möglichkeit des Beschreibungsmechanismus erweitert wird, d. h. beantworten Sie die Frage, ob durch die Erweiterung Sprachen erkannt werden können, die ohne die Erweiterung nicht erkannt werden können. Begründen Sie Ihre Antwort entweder dadurch, dass Sie eine solche Sprache angeben oder einen Mechanismus skizzieren, der die Simulation von Turingmaschinen mit mehreren Spuren auf eine Turingmaschine mit einer Spur durchführt.

Thema Nr. 2**Aufgabe 1:**  
**(formale Sprache)**

Sei  $L$  die Menge aller Worte über dem Alphabet  $\{a, b\}$ ,

deren zweites Zeichen (von links) ein  $a$ ,  
deren drittletztes Zeichen ein  $b$  ist und  
bei denen nie zwei  $a$ 's aufeinander folgen.

Ist  $L$  regulär oder kontext-frei und nicht regulär oder nicht kontextfrei?  
Begründen Sie Ihre Antwort durch die Angabe einer passenden Grammatik oder eines Automaten.

**Aufgabe 2:**  
**(Abschlusseigenschaften)**

Zeigen Sie:

- i) Die regulären Sprachen sind abgeschlossen unter Durchschnitt.
- ii) Die kontextfreien Sprachen sind nicht abgeschlossen unter Durchschnitt.

Begründen Sie Ihre Antwort bei Teilaufgabe i durch die Konstruktion eines entsprechenden Automaten bzw. einer Grammatik ohne den Beweis der Korrektheit ihrer Konstruktion.

**Aufgabe 3:**  
**(kontextfreie Sprachen)**

Definition:

Eine kontextfreie Grammatik  $G = (N, T, P, S)$ , die nur Worte der Länge  $\geq 3$  erzeugt, ist in 3-Normalform, wenn die Produktionen die Form

$$A \rightarrow BCB \quad \text{oder} \quad A \rightarrow a \quad \text{haben}$$

mit nicht-terminalen  $A, B, C, D$  und terminalen Zeichen  $a$ .  
(Die 2-Normalform ist die Chomsky Normalform.)

Zeigen Sie:

Zu jeder kontextfreien Sprache  $L$ , die nur Worte  $x$  der Länge  $\geq 3$  enthält, gibt es eine kontextfreie Grammatik  $G$  in 3-Normalform, die  $L$  erzeugt.

Geben Sie die Konstruktion an. Der Äquivalenz- oder Korrektheitsbeweis ist nicht verlangt. Hinweis: Sie dürfen die Chomsky Normalform verwenden.

#### Aufgabe 4 (NP-Vollständigkeit)

Das  $k$ -Färbbarkeitsproblem auf Graphen ist das Problem, ob man die Knoten eines Graphen  $G$  so mit höchstens  $k$  Farben färben kann, dass benachbarte Knoten verschiedene Farben haben.

Warum ist das  $k$ -Färbbarkeitsproblem in NP?

(Es ist sogar bekannt, dass das  $k$ -Färbbarkeitsproblem NP-vollständig ist.)

#### Aufgabe 5 (Berechenbarkeit)

Was halten Sie von der These:

*„Alles was exakt beschrieben werden kann, kann man auch berechnen.“*

Geben Sie Argumente für oder gegen diese These an.