
Prüfungsteilnehmer	Prüfungstermin	Einzelprüfungsnummer
--------------------	----------------	----------------------

Kennzahl: _____

Kennwort: _____

**Herbst
2014**

66115

Arbeitsplatz-Nr.: _____

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

Fach: **Informatik (vertieft studiert)**

Einzelprüfung: **Theoretische Informatik, Algorithmen**

Anzahl der gestellten Themen (Aufgaben): 2

Anzahl der Druckseiten dieser Vorlage: 6

Bitte wenden!

Thema Nr. 1
(Aufgabengruppe)

Es sind alle Aufgaben dieser Aufgabengruppe zu bearbeiten!

Aufgabe 1:

Wir fixieren das Alphabet $\Sigma = \{0, 1\}$ und definieren $L \subseteq \Sigma^*$ durch

$$L = \{w \mid \text{in } w \text{ kommt das Teilwort } 0010 \text{ höchstens einmal vor}\}$$

Es ist also $000 \in L$, $00010 \in L$, aber $00010000100 \notin L$ und auch $0010010 \notin L$.

- a) Zeigen Sie, dass L regulär ist.
- b) Vervollständigen Sie durch Hinzufügen eines weiteren Zustandes, sowie von Kanten und der Angabe der Endzustände folgendes Diagramm zu einem deterministischen Automaten für L . Übertragen Sie dazu das Diagramm auf Ihr Arbeitspapier.

$$\rightarrow q_0 \xrightarrow{0} q_1 \xrightarrow{0} q_2 \xrightarrow{1} q_3 \xrightarrow{0} q_4 \xrightarrow{0} q_5 \xrightarrow{1} q_6 \xrightarrow{0} q_7$$

- c) Zeigen Sie durch Ausführung des Minimierungsalgorithmus, dass dieser Automat minimal ist.
- d) Geben Sie die Äquivalenzklassen der Myhill-Nerode Äquivalenz von L durch Repräsentanten an. (Diese Äquivalenz ist definiert durch $x \sim_L y \iff \forall u. xu \in L \iff yu \in L$.)

Geben Sie zu zwei Klassen Ihrer Wahl neben dem gewählten Repräsentanten noch ein weiteres Element an.

Geben Sie für die Klasse, in der sich das leere Wort befindet, einen regulären Ausdruck an.

Aufgabe 2:

Ordnen Sie die folgenden formalen Sprachen bestmöglich in die Chomsky-Hierarchie ein und geben Sie eine ausreichende Begründung an:

- a) $L_1 = \{(ab)^n a (ab)^n \mid n \geq 1\}$
- b) $L_2 = \{(ab)^n (ab)^n \mid n \geq 1\}$
- c) $L_3 = \{(ab)^n (ab)^n (ab)^n \mid n \geq 1\}$
- d) $L_4 = \{(ab)^n (ab)^n a (ab)^n \mid n \geq 1\}$
- e) $L_5 = \{(ab)^n a (ab)^n a (ab)^n \mid n \geq 1\}$. (Bei dieser Teilaufgabe ist keine Begründung notwendig.)
- f) L_6 sei die Menge aller syntaktisch korrekten Java-Programme, die ohne Eingabe nicht terminieren.
- g) L_7 sei die Menge aller wohlgeformten Klammerausdrücke, also $() , (()) , ((())) \in L_7$, aber $) , ())(, (((() \notin L_7$.

Fortsetzung nächste Seite!

Aufgabe 3:

Bekanntlich sind beim RUCKSACK-Problem natürliche Zahlen a_1, \dots, a_n und eine Zielzahl b gegeben und es ist gefragt, ob eine Teilmenge $I \subseteq \{1, \dots, n\}$ existiert, sodass $\sum_{i \in I} a_i = b$. Dieses Problem ist bekanntermaßen NP-vollständig.

- a) Zeigen Sie, dass die Variante GRUCKSACK, bei der die Zahlen a_i und b allesamt gerade sein müssen, ebenfalls NP-vollständig ist.
- b) Zeigen Sie, dass die Variante BZRUCKSACK, bei der die Zahl b eine Zweierpotenz sein muss, ebenfalls NP-vollständig ist.
- c) In welche Komplexitätsklasse fällt die Variante ZARUCKSACK, bei der es nur zwei verschiedene Arten von Gewichten gibt, also zwei Zahlen u, v existieren, sodass $a_i \in \{u, v\}$ für $i = 1, \dots, n$?

Aufgabe 4:

Es seien natürliche Zahlen a_1, \dots, a_n in Form eines Arrays gegeben. Die Zahlen a_i steigen bis zu einem gewissen Punkt an und fallen anschließend wieder. Formal existiert also $m \leq n$, sodass Folgendes gilt: Ist $1 \leq i < m$, so ist $a_i < a_{i+1}$. Ist $m \leq i < n$, so ist $a_i > a_{i+1}$.

- a) Entwerfen Sie einen Algorithmus, der den größten Eintrag des Arrays, also $\max_i a_i$ in Zeit $O(\log n)$ bestimmt. Genügt die Eingabe nicht der Spezifikation, so darf sich Ihr Algorithmus völlig beliebig verhalten.
- b) Jetzt erlauben wir auch die Möglichkeit, dass die Arraywerte "liegenbleiben", also dass $m < n$ existiert, sodass gilt: Ist $1 \leq i < m$, dann ist $a_i \leq a_{i+1}$. Ist $m \leq i < n$, dann ist $a_i \geq a_{i+1}$.

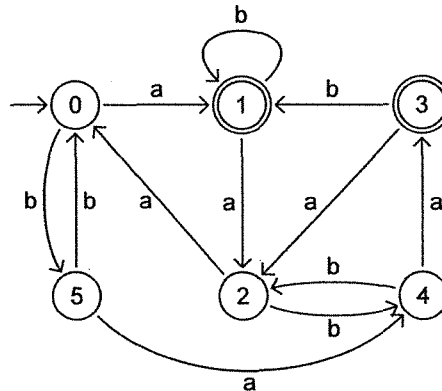
Begründen Sie, dass kein Algorithmus mit Laufzeit $O(\log n)$ existiert, der entscheidet, ob ein Array von der oben beschriebenen Form ist. Hilfe: Lassen Sie einen hypothetischen Algorithmus auf der Eingabe $a_i = 0$ für $i = 1, \dots, n$ laufen.

Thema Nr. 2
(Aufgabengruppe)

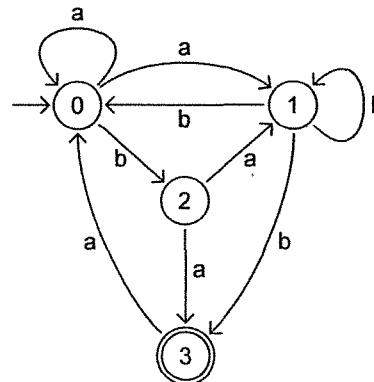
Es sind alle Aufgaben dieser Aufgabengruppe zu bearbeiten!

Aufgabe 1:

- a) Minimieren Sie folgenden deterministischen, endlichen Automaten.



- b) Bestimmen Sie einen regulären Ausdruck für die von dem folgenden nichtdeterministischen, endlichen Automaten akzeptierte Sprache.



Aufgabe 2:

Beweisen Sie folgende Aussagen.

- a) $L_1 = \{w \in \{0, 1\}^* \mid \text{die Länge von } w \text{ ist durch 2 oder 3 teilbar}\}$ ist regulär.
 b) $L_2 = \{w \in \{0, 1\}^* \mid w = 0^i 1^j 0^k \text{ mit } i, j, k \in \mathbb{N} \text{ und } j = i + k\}$ ist kontextfrei.

Fortsetzung nächste Seite!

Aufgabe 3:

- a) Definieren Sie eine berechenbare Funktion $f: \mathbb{N} \rightarrow \mathbb{N}$ mit entscheidbarem Definitionsbereich und unentscheidbarem Wertebereich. Begründen Sie Ihre Aussagen.
- b) Beweisen oder widerlegen Sie folgende Aussagen.
- Jede berechenbare Funktion $h: \mathbb{N} \rightarrow \mathbb{N}$ mit endlichem Wertebereich besitzt einen entscheidbaren Definitionsbereich.
 - Jede berechenbare Funktion $h: \mathbb{N} \rightarrow \mathbb{N}$ mit endlichem Definitionsbereich besitzt einen entscheidbaren Wertebereich.

Aufgabe 4:

- a) Die Fibonacci-Funktion $fib: \mathbb{N} \rightarrow \mathbb{N}$ ist definiert durch

$$fib(x) = \begin{cases} x, & \text{falls } x \in \{0, 1\} \\ f(x-1) + f(x-2) & \text{sonst.} \end{cases}$$

Beweisen Sie, dass fib nicht in polynomieller Zeit berechenbar ist.

- b) Sei $f: \mathbb{N} \rightarrow \mathbb{N}$ eine totale, in polynomieller Zeit berechenbare Funktion mit $f(x) \geq x$ für alle $x \in \mathbb{N}$. W_f bezeichne den Wertebereich der Funktion f . Beweisen Sie, dass $W_f \in NP$.

Aufgabe 5:

Gegeben sei eine Standarddatenstruktur Stapel (Stack) mit den Operationen

- void Push(Element e),
- Element Pop(),
- Boolean Empty().

sowie dem Standardkonstruktor Stapel(), der einen leeren Stapel zur Verfügung stellt.

- a) Geben Sie eine Methode Stapel Merge(Stack S , Stack T) an, die einen aufsteigend geordneten Stapel zurückgibt, unter der Bedingung, dass die beiden übergebenen Stapel aufsteigend sortiert sind, d.h. $S.Pop()$ liefert das größte Element in S zurück und $T.Pop()$ liefert das größte Element in T zurück. Als Hilfsdatenstruktur dürfen Sie nur Stapel verwenden, keine Felder oder Listen.
- b) Analysieren Sie die Laufzeit Ihrer Methode.

Aufgabe 6:

Gegeben sei ein einfacher Sortieralgorithmus, der ein gegebenes Feld A dadurch sortiert, dass er das Minimum m von A findet, dann das Minimum von A ohne das Element m usw.

- a) Geben Sie den Algorithmus in Pseudocode an. Implementieren Sie den Algorithmus *in situ*, d.h. so, dass er außer dem Eingabefeld nur konstanten Extraspeicher benötigt.
- b) Analysieren Sie die Laufzeit Ihres Algorithmus.
- c) Geben Sie eine Datenstruktur an, mit der Sie Ihren Algorithmus beschleunigen können.

Fortsetzung nächste Seite!

Aufgabe 7:

Gegeben sei folgende Adjazenzmatrix des Graphen G („ ∞ “ bedeutet, dass zwischen den beiden Knoten keine direkte Verbindung existiert):

	Q	A	B	C	D	E	F
Q	0	1	3	∞	5	∞	∞
A	1	0	∞	∞	3	1	∞
B	3	∞	0	4	1	∞	∞
C	∞	∞	4	0	2	∞	2
D	5	3	1	2	0	3	1
E	∞	1	∞	∞	3	0	8
F	∞	∞	∞	2	1	8	0

- a) Bestimmen Sie mit dem Algorithmus von *Dijkstra* den kürzesten Weg von der Quelle Q bis zu den Senken C und F. Verwenden Sie zur Lösung eine Tabelle nach unten stehendem Muster, markieren Sie in jeder Zeile den jeweils als nächstes zu betrachtenden Knoten und führen Sie die Prioritätsliste der noch zu betrachtenden Knoten in der letzten Spalte (der nächste Knoten steht links):

Q	A	B	C	D	E	F	Queue
0	∞	∞	∞	∞	∞	∞	Q

- b) Geben Sie den vorangehend ermittelten, kürzesten Weg von Q zu F als Knotenfolge an.
- c) Bestimmen Sie mit Hilfe des Verfahrens nach *Kruskal* den kleinsten Spannbaum des Graphen G . Geben Sie die Kanten (Knotenpaare) in der Reihenfolge an, in der Sie sie dem minimalen Spannbaum gemäß Kruskal hinzufügen würden. Wie groß ist die Kantengewichtssumme im Spannbaum insgesamt?