Prüfungsteilnehmer	Prüfungstermin	Einzelprüfungsnummer
Kennzahl: ______ Kennwort: ______ Arbeitsplatz-Nr.: ______	**Frühjahr 2014**	**46115**

Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen
— Prüfungsaufgaben —

Fach: **Informatik (Unterrichtsfach)**

Einzelprüfung: **Th. Informatik, Algorith./Datenstr.**

Anzahl der gestellten Themen (Aufgaben): **2**

Anzahl der Druckseiten dieser Vorlage: **6**

Bitte wenden!

Thema Nr. 1

Annahmen:
Sie dürfen als bekannt und bewiesen voraussetzen:
Die Sprache $\{a^n b^n \mid n \geq 1\}$ ist nicht regulär
Die Sprache $\{a^n b^n c^n \mid n \geq 1\}$ ist nicht kontext-frei.
Um zu zeigen, dass eine Sprache L regulär (kontexfrei) ist, reicht die Angabe einer entsprechenden Beschreibung (Automat, Grammatik, Ausdruck).
Sie müssen nicht mehr zeigen, dass Ihre Beschreibung korrekt ist und genau die vorgegebene Sprache beschreibt.

Aufgabe 1: Reguläre Mengen

Sei L die Menge aller Wörter über dem Alphabet $\{a,b\}$,
bei denen das erste, zweite und letzte Zeichen übereinstimmen.

a) Geben Sie alle Worte bis zur (einschließlich) Länge drei an!
b) Beschreiben Sie L
   i) durch einen regulären Ausdruck
   ii) durch einen deterministischen endlichen Automaten A!

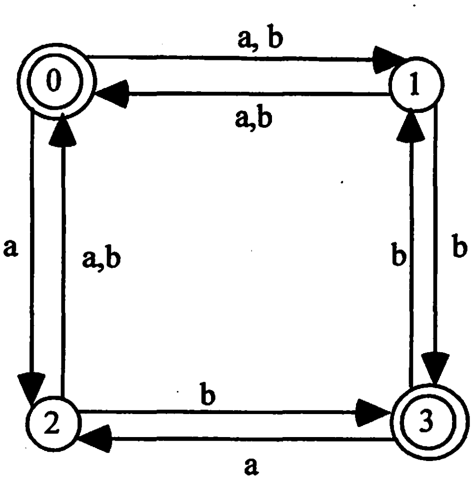
Aufgabe 2: Regulär oder nicht

Sei L die Sprache $L = \{ a^n b^m \mid m = n^2, n \geq 1\}$

Ist L regulär oder nicht?
Begründen Sie Ihre Antwort durch die Angabe einer passenden Beschreibung für L oder den Nachweis, dass L nicht regulär sein kann!

Aufgabe 3: NFA und DFA

Konstruieren Sie zum angegebenen nicht-deterministischen Automaten
$A = (\{0,1,2\}, \{a,b\}, \delta, 0, \{0,3\})$
einen äquivalenten deterministischen endlichen Automaten!
A hat die Zustände 0,1,2,3; 0 ist der Startzustand und 0,3 sind Endzustände.



Fortsetzung nächste Seite!

Aufgabe 4: Kontextfrei

Zeigen Sie, dass die Sprache L kontext-frei ist!

$L = \{ a^n b^k c^k b^m c^m a^n \mid k, m, n \geq 1\}$

Aufgabe 5: Berechenbarkeit und Komplexität

Gegeben sei die Sprache $L = \{ a^j b^m c^k \mid j, m, k \geq 1, j \neq m \text{ und } j \neq k \}$

a) Geben Sie eine Turingmaschine M an, die L erkennt!
Beschreiben Sie in Worten, wie Ihre Turingmaschine arbeitet!

b) Welche Zeitkomplexität in O-Notation hat Ihre Turingmaschine?
Erläutern Sie dies anhand Ihrer in a) gegebenen Beschreibung!

Aufgabe 6: O-Notation

Gegeben sind die Funktionen f und g (über den natürlichen Zahlen):

$$g(n) = 100(\sqrt{n}+1)^2 \text{ und } f(n) = n^3 - 2n + 3$$

Zeigen Sie dass $g \in O(f)$ gilt!
(Es reicht nicht zu sagen, dass eine Funktion stärker steigt).

Aufgabe 7: Heap und binärer Suchbaum und AVL Baum

a) Fügen Sie nacheinander die Zahlen 11, 1, 2, 13, 9, 10, 7, 5

(i) in einen leeren binären Suchbaum
und zeichnen Sie den Suchbaum jeweils nach dem Einfügen von „9“ und „5“

(ii) in einen leeren Min-Heap ein, der bzgl. „≤“ angeordnet ist
und geben Sie den Heap nach „9“ und nach „5“ an

(iii) in einen leeren AVL- Baum ein!
Geben Sie den AVL Baum nach „2“ und „5“ an und
beschreiben Sie die ggf. notwendigen Rotationen beim Einfügen dieser beiden Elemente!

Fortsetzung nächste Seite!

Aufgabe 8: Minimaler Spannbaum

Bestimmen Sie einen minimalen Spannbaum für einen ungerichteten Graphen, der durch die nachfolgende Entfernungsmatrix gegeben ist!
Die Matrix ist symmetrisch und „∞" bedeutet, dass es keine Kante gibt.

Geben Sie die Spannbaumkanten ein!
Wie groß (Wert) ist der minimale Spannbaum?

	A	B	C	D	E	F	G	H
A	0	8	–1	∞	8	∞	7	∞
B	8	0	∞	2	∞	∞	∞	9
C	–1	∞	0	5	8	1	7	∞
D	∞	2	5	0	6	6	∞	∞
E	8	∞	8	6	0	6	3	∞
F	∞	∞	1	6	6	0	11	4
G	7	∞	7	∞	3	11	0	5
H	∞	9	∞	∞	∞	4	5	0

Thema Nr. 2

Aufgabe 1:

Sei $\Sigma = \{0, 1, \$\}$ und sei $\mathbb{N}_0 = \{0, 1, 2, \ldots\}$.

a) Sei

$$L_1 = \{0^n\$1^{2n} \mid n \in \mathbb{N}_0\} \cup \{0^{2n}\$1^n \mid n \in \mathbb{N}_0\} \ .$$

Beispiele: $00\$1111 \in L_1$, $\$ \in L_1$, $00\$1 \in L_1$.

a1) Zeigen Sie, dass L_1 kontextfrei ist, indem Sie eine kontextfreie Grammatik G angeben mit $L(G) = L_1$, und
a2) begründen Sie, warum Ihre Grammatik *genau* die Sprache L_1 erzeugt.

b) Formulieren Sie das Pumping-Lemma für reguläre Sprachen:

„*Sei L eine reguläre Sprache über dem Alphabet* Σ. *Dann gibt es ...*"

c) Zeigen Sie mittels Pumping-Lemma für reguläre Sprachen, dass die Sprache L_1 nicht regulär ist.

Aufgabe 2:

a) Geben Sie einen deterministischen endlichen Automaten an (Zeichnung), der die Sprache

$$L_2 = \{a^i \mid i \in \mathbb{N},\ i \text{ ist durch 2, aber nicht durch 3 teilbar}\}$$

akzeptiert. ($\mathbb{N} = \{1, 2, \ldots\}$)

b) Beweisen oder widerlegen Sie die folgende Behauptung:

„*Ist die Sprache L nicht regulär, dann ist auch jede echte Teilmenge von L nicht regulär.*"

c) Für $n \geq 2$ sei zu einem Wort $w = a_1 a_2 \ldots a_n$:

$$\text{kopflos}(w) = \{a_2 \ldots a_n\} \ .$$

Außerdem sei $\text{kopflos}(a) = \{\varepsilon\}$ und $\text{kopflos}(\varepsilon) = \emptyset$.

Beweisen oder widerlegen Sie die folgende Behauptung:

„*Ist die Sprache L regulär, dann ist auch*

$$kopflos(L) = \bigcup_{w \in L} kopflos(w)$$

regulär."

Fortsetzung nächste Seite!

Aufgabe 3:

a) Fügen Sie die Zahlen 17, 7, 21, 3, 10, 13, 1, 5 nacheinander in der vorgegebenen Reihenfolge in einen binären Suchbaum ein und zeichnen Sie das Ergebnis!

b) Implementieren Sie in einer objektorientierten Programmiersprache oder in entsprechendem Pseudocode eine rekursiv festgelegte Datenstruktur, deren Gestaltung sich an folgender Definition eines binären Baumes orientiert!

> *Ein binärer Baum ist entweder ein leerer Baum oder besteht aus einem Wurzelelement, das einen binären Baum als linken und einen als rechten Teilbaum besitzt.*

Bei dieser Teilaufgabe können Sie auf die Implementierung von Methoden (außer ggf. notwendigen Konstruktoren) verzichten!

c) Beschreiben Sie durch Implementierung in einer gängigen objektorientierten Programmiersprache oder in entsprechendem Pseudocode, wie bei Verwendung der obigen Datenstruktur die Methode *loescheKnoten(w)* gestaltet sein muss, mit der der Knoten mit dem Eintrag *w* aus dem Baum entfernt werden kann, ohne die Suchbaumeigenschaft zu verletzen!

Aufgabe 4:

Für Binomialkoeffizienten $\binom{n}{k}$ gelten neben den grundlegenden Beziehungen $\binom{n}{0} = 1$ und $\binom{n}{n} = 1$ auch folgende Formeln:

$$\text{A)} \binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}$$

$$\text{B)} \binom{n}{k} = \binom{n-1}{k-1} \cdot \frac{n}{k}$$

a) Implementieren Sie unter Verwendung von Beziehung A) eine rekursive Methode ***binRek(n,k)*** zur Berechnung des Binomialkoeffizienten in einer objektorientierten Programmiersprache oder entsprechendem Pseudocode!

b) Implementieren Sie unter Verwendung von Beziehung B) eine iterative Methode ***binIT(n,k)*** zur Berechnung des Binomialkoeffizienten in einer objektorientierten Programmiersprache oder entsprechendem Pseudocode!

c) Geben Sie die Laufzeitkomplexität der Methoden ***binRek(n,k)*** und ***binIT(n,k)*** aus den vorhergehenden beiden Teilaufgaben in O-Notation an!