Lehrstuhl für Softwaretechnik und Programmiersprachen
Professor Dr. Michael Leuschel

HEINRICH HEINE
UNIVERSITÄT DÜSSELDORF

Grundlagen der Theoretischen Informatik - Sommersemester 2014

Klausur

Klausurtermin: 18. Juli 2014

- **BITTE NICHT MIT BLEISTIFT ODER ROTSTIFT SCHREIBEN!**
- **TRAGEN SIE AUF JEDEM BLATT IHREN NAMEN UND MATRIKELNUMMER EIN!**
- **BEI ANGABE VON MEHREREN LÖSUNGEN WIRD STETS DIE LÖSUNG MIT DER GERINGEREN PUNKTZAHL GEWERTET!**

Name, Vorname:

Studienfach, Semester:

Matrikelnummer:

Aufgabe	1	2	3	4	5	Gesamt
erreichbare Punktzahl	20	15	20	25	10	90
erreichte Punktzahl						

Erlaubte Hilfsmittel:

- Vorlesungsskript, Übungsblätter,
- Bücher, Vorlesungs- und Übungsmitschriften.

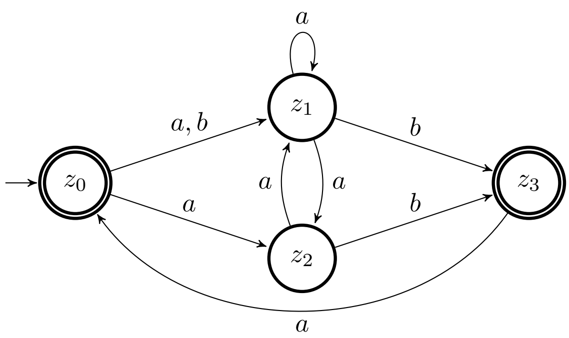
Nicht erlaubte Hilfsmittel:

- Elektronische Geräte,
- Kommiliton/inn/en.

Achten Sie darauf, dass Rechenwege und Zwischenschritte vollständig und ersichtlich sind.

Name: **Matrikelnummer:**

Aufgabe 1 (20 Punkte)
Sei folgender NFA $M = (\{a, b\}, \{z_0, z_1, z_2, z_3\}, \delta, z_0, \{z_0, z_3\})$ gegeben:



(a) [12 Punkte] Konstruieren Sie zu M einen äquivalenten minimalen DFA N, so dass $L(M) = L(N)$ gilt.

(b) [8 Punkte] Geben Sie einen regulären Ausdruck γ an, so dass $L(M) = L(\gamma)$ gilt.

(Bitte geben Sie alle Zwischenschritte der Rechnungen in allen Teilaufgaben an!)

Name: **Matrikelnummer:**

Name: **Matrikelnummer:**

Aufgabe 2 (15 Punkte)
Gegeben sei die kontextfreie Grammatik $G = (\{a, b\}, \{S, A, B, C, D\}, S, P)$ mit folgender Regelmenge:

$$
\begin{aligned}
P \;=\; \{ \quad S &\rightarrow \; aaD \mid bA, \\
A &\rightarrow \; C, \\
B &\rightarrow \; \lambda, \\
C &\rightarrow \; D \mid AB, \\
D &\rightarrow \; bD \mid a \quad \}
\end{aligned}
$$

(a) [8 Punkte] Überführen Sie die Grammatik G in eine Grammatik $G^{'}$ in Chomsky-Normalform.

Hinweis: Bitte geben Sie alle Zwischenschritte für die Überführung in CNF an. Es gibt Punkteabzüge für nicht angegebene Zwischenschritte.

(b) [7 Punkte] Zeigen Sie, dass das Wort *bbbba* in $L(G^{'})$ liegt, indem Sie die untere Tabelle anhand des CYK-Algorithmus ausfüllen. Dabei müssen Sie die in Teil (a) konstruierte Grammatik $G^{'}$, die in CNF ist, benutzen und alle Zwischenschritte in der Tabelle angeben.

i	1	2	3	4	5
4					
3					
2					
1					
0					
j	b	b	b	b	a

(Bitte geben Sie alle Zwischenschritte der Rechnungen in allen Teilaufgaben an!)

Name: **Matrikelnummer:**

Aufgabe 3 (20 Punkte)
Gegeben sei die folgende Sprache L:

$$L = \{0^i 1^j 2^k \mid (i, j, k \geq 1) \wedge (i \leq j \vee j \leq k)\}.$$

(a) [12 Punkte] Zeigen Sie, entweder durch Angabe einer kontextfreien Grammatik oder durch Angabe eines nicht-deterministischen Kellerautomaten, dass L eine kontextfreie Sprache ist.

(b) [8 Punkte] Zeigen Sie, dass L keine reguläre Sprache ist.

(Bitte geben Sie alle Zwischenschritte der Rechnungen in allen Teilaufgaben an!)

Name: **Matrikelnummer:**

Aufgabe 4 (25 Punkte)
Sei Σ ein Alphabet. Die Wortfunktion $FIRST_k : \Sigma^* \to \Sigma^*$, die für $k \geq 0$ durch

$$FIRST_k(x) = \begin{cases} x, & \text{falls } \mid x \mid \leq k \\ y, & \text{falls } x = yz, \ \mid y \mid = k, \text{ für } y, z \in \Sigma^*. \end{cases}$$

definiert ist, ist für jedes feste k Turing-berechenbar. D.h. man kann für jedes $k \geq 0$ eine Turingmaschine $M_k = (\Sigma, \Gamma, Z, \delta_k, z_0, \Box, F)$ angeben, so dass für alle $x, y \in \Sigma^*$ die folgende Äquivalenz erfüllt ist:

$$FIRST_k(x) = y \Leftrightarrow z_0 x \vdash^*_{M_k} zy \text{ für ein } z \in F.$$

Man kann formal die Übergangsfunktion δ_k der Turingmaschine M_k wie folgt definieren:

$$\begin{aligned}
(z_0, \Box) &\mapsto (z_e, \Box, N) & \\
(z_0, a) &\mapsto (z_1, a, R) & \forall a \in \Sigma \\
(z_1, \Box) &\mapsto (z', \Box, L) & \\
(z_1, a) &\mapsto (z_2, a, R) & \forall a \in \Sigma \\
(z_2, \Box) &\mapsto (z', \Box, L) & \\
(z_2, a) &\mapsto (z_3, a, R) & \forall a \in \Sigma \\
&\vdots & \\
(z_{k-1}, \Box) &\mapsto (z', \Box, L) & \\
(z_{k-1}, a) &\mapsto (z, a, R) & \forall a \in \Sigma \\
(z, \Box) &\mapsto (z_l, \Box, L) & \\
(z, a) &\mapsto (z, \Box, R) & \forall a \in \Sigma \\
(z_l, \Box) &\mapsto (z_l, \Box, L) & \\
(z_l, a) &\mapsto (z', a, L) & \forall a \in \Sigma \\
(z', \Box) &\mapsto (z_e, \Box, R) & \\
(z', a) &\mapsto (z', a, L) & \forall a \in \Sigma
\end{aligned}$$

Dabei sind $Z = \{z_0, z_1, \ldots, z_{k-1}, z_l, z, z', z_e\}$ und $F = \{z_e\}$.

(a) [10 Punkte] Sei $\Sigma = \{0, 1\}$ und sei $k = 3$. Geben Sie die Konfigurationenfolgen der Turingmaschine M_3 für die Wörter 0101 und 01 an.

(b) [15 Punkte] Zeigen Sie, dass die Wortfunktion $LAST_k : \Sigma^* \to \Sigma^*$ mit

$$LAST_k(x) = \begin{cases} x, & \text{falls } \mid x \mid \leq k \\ z, & \text{falls } x = yz, \ \mid z \mid = k, \text{ für } y, z \in \Sigma^*. \end{cases}$$

für jedes feste $k \geq 0$ Turing-berechnbar ist. D.h. geben Sie die allgemeine Konstruktion (ähnlich wie oben) für jede Turingmaschine $N_k = (\Sigma, \Gamma, Z', \delta'_k, z_0, \Box, F')$ an, die die folgende Bedingung für alle $x, y \in \Sigma^*$ erfüllt:

$$LAST_k(x) = y \Leftrightarrow z_0 x \vdash^*_{N_k} zy \text{ für ein } z \in F'.$$

(Bitte geben Sie alle Zwischenschritte der Rechnungen in allen Teilaufgaben an!)

Name: **Matrikelnummer:**

Aufgabe 5 (10 Punkte)

(a) [5 Punkte] Sei $f : \mathbb{N} \to \mathbb{N}$ eine Funktion, die durch das folgende WHILE-Programm berechnet wird:

$x_0 := 0;\ x_2 := x_1 + 0;\ x_3 := 1;$
WHILE $x_2 \neq 0$ DO
  LOOP x_3 DO
    $x_0 := x_0 + 1;$
  END;
  $x_2 := x_2 - 1;$
  $x_3 := x_3 + 1;$
END;

Das WHILE-Programm startet mit n in der Variable x_1 und stoppt mit dem Wert $f(n)$ in der Variable x_0, wobei $n \in \mathbb{N}$.

Geben Sie eine formale mathematische Beschreibung für die Funktion f an.

(b) [5 Punkte] Gegeben sei die partiell rekursive Funktion $\overline{xor} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, die wie folgt definiert ist:

$$\overline{xor}(x, y) = \begin{cases} 0, & \text{falls } x = 0 \text{ und } y = 1, \text{ oder } x = 1 \text{ und } y = 0 \\ 1, & \text{falls } x = y = 0, \text{ oder } x = y = 1 \\ \text{undefiniert}, & \text{falls } x > 1 \text{ oder } y > 1 \end{cases}$$

Welche Funktion berechnet $\mu\ \overline{xor}$? Bitte geben Sie eine formale mathematische Beschreibung für $\mu\ \overline{xor}$ an.

(Bitte geben Sie alle Zwischenschritte der Rechnungen in allen Teilaufgaben an!)