Klausur im SS 22, 19.07.2022

Theoretische Informatik für Angewandte Informatik

Prof. Dr. Barbara Staehle, HTWG Konstanz

Bearbeitungszeit: 60 Minuten

Hinweise:

- Falls Sie für die Aufgaben alle Punkte haben wollen, begründen Sie Ihre Antworten, bzw. stellen Sie den Lösungs- / Rechenweg nachvollziehbar dar.
- Lösen Sie sofern möglich, die Aufgaben auf dem Angabenblatt. Falls nicht genügend Platz vorhanden ist, nutzen Sie zusätzliches Papier.
- **Die Klausur enthält mehr Aufgaben, als Sie in der Bearbeitungszeit lösen können. Wählen Sie klug aus welche Aufgaben Sie lösen!** Sie müssen weder zum Bestehen noch für eine sehr gute Note alle Aufgaben korrekt bearbeiten. Zum Bestehen reichen ca. **50 Punkte**, eine sehr gute Note gibt es **ab ca. 90** Punkten.

Name: ______________________

Matrikelnummer: ____________

Note: ____________

Aufgabe	1	2	3	4	5	Σ
Punkte	20	22	19	25	24	110
Erreicht						

Aufgabe 1, 20 Punkte
Wahr oder Falsch?

Sind folgenden Aussagen wahr oder falsch? Begründen Sie Ihre Entscheidung (kurz).

Punktvergabe: w/f richtig: 1 Punkt; w/f richtig und Begründung sinnvoll: 2 Punkte

Aussage	wahr	falsch	kurze Begründung
(a) $L_a = \{ain, win\}$ ist eine formale Sprache vom Chomsky-Typ 3 über dem Alphabet $\Sigma_a = \{a, b, \ldots, z\}$.	☐	☐	
(b) Es gilt P ⊂ NP ⊂ PSPACE = NPSPACE ⊂ EXP ⊂ NEXP	☐	☐	
(c) Wenn man von einer beliebigen Menge das Komplement bildet und dann den Durchschnitt von diesem mit der Ausgangsmenge bildet, ergibt sich die leere Menge.	☐	☐	
(d) Alle Algorithmen, die das selbe Problem lösen, liegen in der selben Zeitkomplexitätsklasse.	☐	☐	
(e) Zukünftige Compiler werden zuverlässig für jedes Programm erkennen können, ob dieses in eine Endlosschleife läuft.	☐	☐	
(f) Jede reguläre Sprache kann auch von einer kontextfreien Grammatik erzeugt werden.	☐	☐	
(g) Ein DEA kann nicht erkennen, ob ein String zu jeder öffnenden Klammer auch eine schließende enthält.	☐	☐	
(h) Ein Kellerautomat (PDA oder DPDA) kann in einem Schritt beliebig viele Zeichen aus dem Keller lesen und in den Keller schreiben.	☐	☐	
(i) Wird aus einer Grammatik mit X verschiedenen Nonterminalsymbolen ein Wort der Länge $l > X$ abgeleitet, muss mindestens ein Nonterminal in der Ableitung mehrfach vorkommen.	☐	☐	
(j) Endliche Automaten halten immer an. Kellerautomaten und Turing-Maschinen können auch unendlich lange laufen	☐	☐	

AUFGABE 2, 22 PUNKTE
LOGIK UND FORMALE SPRACHEN

2.1 LOGIK

Folgendes sei gegeben:

- das Alphabet $\Sigma_0 = \{x, y\}$
- die Menge $R = \{r_1, r_2, r_3\}$ mit den regulären Ausdrücken

$$r_1 = [xy]^+, \quad r_2 = (xy)^*, \quad r_3 = x^* y^*$$

- die Menge $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$ mit den Worten

$$\omega_1 = xyxyxy, \quad \omega_2 = yxyxy, \quad \omega_3 = xyy, \quad \omega_4 = \varepsilon$$

- Aussageform $E(\omega, r)$: „Wort ω wird vom regulären Ausdruck r erzeugt: $\omega \in \mathscr{L}(r)$“ (für $\omega \in \omega, r \in R$)
- Aussageform $D(M, N)$: „M und N sind diskjunkt“ (für zwei beliebige Mengen M, N)

Formulieren Sie die folgenden logischen Aussagen in Ihren eigenen Worten als deutsche Sätze **und geben Sie den Wahrheitswert der Aussage an**:

(a) (3 Punkte) $\forall_{\omega\in\Omega}\ \exists_{r\in R}\ E(\omega, r)$

(b) (3 Punkte) $\exists_{r\in R}\ \forall_{\omega\in\Omega}\ E(\omega, r)$

(c) (3 Punkte) $D(\mathscr{L}(r_1), \Sigma_0) \Leftrightarrow D(\mathscr{L}(r_2), \Sigma_0)$

Formulieren Sie folgende Sätze als zusammengesetzte logische Aussagen (bei Bedarf mit Quantoren) **und geben Sie den Wahrheitswert der Aussage an**.

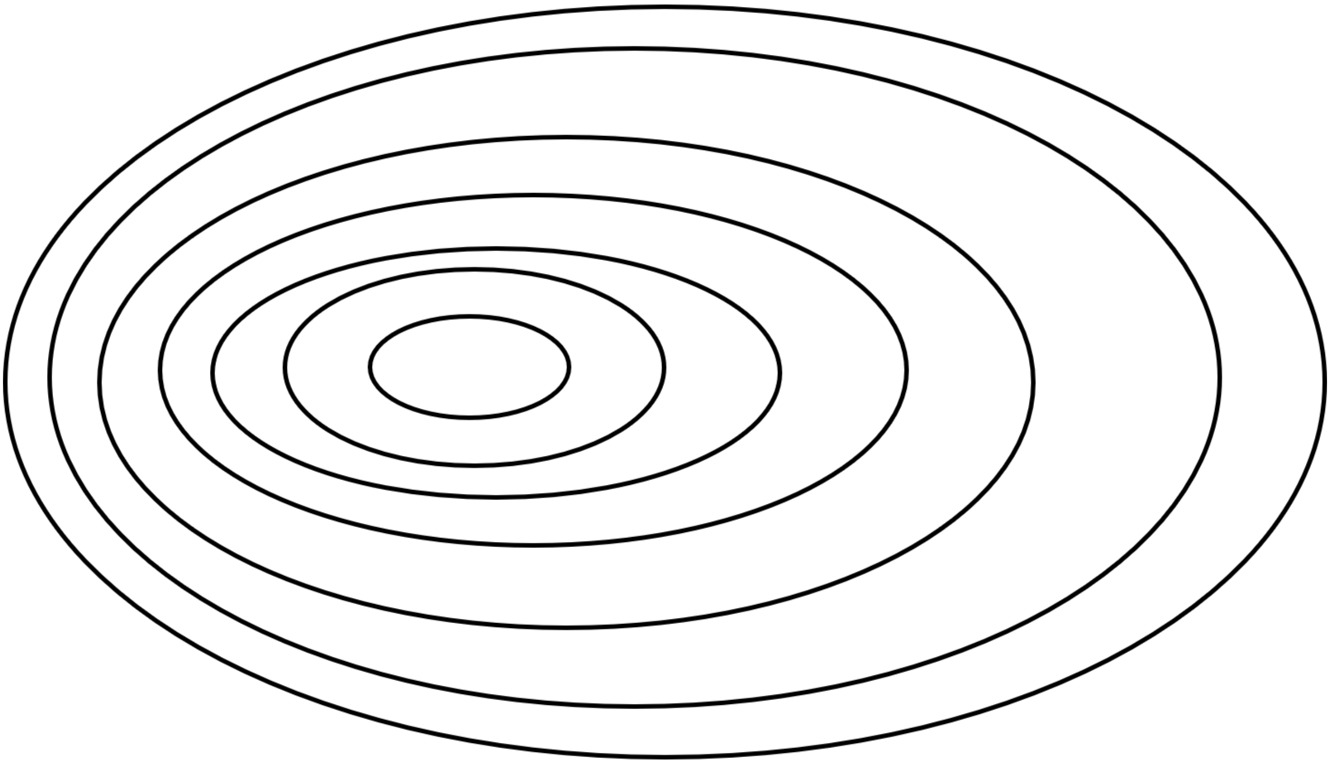
(d) (3 Punkte) Es ist falsch, dass mindestens ein Wort aus Ω existiert, welches von mindestens einem regulären Ausdruck aus R erzeugt wird.

(e) (3 Punkte) Die von je zwei verschiedenen regulären Ausdrücken aus R erzeugten Sprachen sind disjunkt.

2.2 (7 Punkte) Betrachten Sie das unten dargestellte Mengendiagramm. Beschriften Sie das Mengendiagramm so, dass dieses die Ihnen bekannten Inklusionsbeziehungen zwischen den unten stehenden Sprachklassen und Sprachen darstellt. Verwenden Sie für Ihre Beschriftung die in den Klammern angegebenen Abkürzungen.

Die Klassen der

- Chomsky Typ0-, Typ1-, Typ2-, Typ3-Sprachen, $\mathscr{L}_0, \mathscr{L}_1, \mathscr{L}_2, \mathscr{L}_3$
- von den jeweiligen Automatentypen erkannten Sprachen $\mathscr{L}(DEA), \mathscr{L}(NEA), \mathscr{L}(DPDA), \mathscr{L}(PDA), \mathscr{L}(LBA), \mathscr{L}(TM), \mathscr{L}(NTM)$
- entscheidbaren Sprachen (E)
- semi-entscheidbaren Sprachen (SE)
- formalen Sprachen (F)



Hinweis: Manche Ellipsen müssen mehrfach beschriftet werden.

Aufgabe 3, 19 Punkte
Reguläre Sprachen

3.1 Gegeben sei das Alphabet $\Sigma_P = \{f, h, i, l, p, y\}$.
Über diesem ist die formale Sprache $L_P \subseteq \Sigma_P^*$ definiert, welche die Menge der möglichen Schreibweisen des Namens "phillipp" enthält:

- der Name darf mit "f" oder "ph" anfangen (z.B. "phillipp", oder "fillipp")
- die Vokale dürfen jeweils "i" oder "y" sein (z.B. "phylipp", oder "fylyp")
- "l" und "p" können jeweils einmal oder zweimal vorkommen (z.B. "phylipp", oder "filyp", oder "phillipp")

(a) (3 Punkte) Geben Sie den regulären Ausdruck r_P an, der L_P erzeugt.

(b) (6 Punkte) Konstruieren Sie einen endlichen Automaten (deterministisch oder nichtdeterministisch) A_P, der L_P akzeptiert.

(c) (4 Punkte) Geben Sie eine **kontextfreie** Grammatik G_{Pk} an welche L_P erzeugt, für welche also $\mathscr{L}(G_{Pk}) = L_P$ gilt.

(d) (6 Punkte) Geben Sie eine **reguläre** Grammatik G_{Pr} an, welche L_P erzeugt, für welche also $\mathscr{L}(G_{Pr}) = L_P$ gilt.

Aufgabe 4, 25 Punkte
Kontextfreie Sprachen

4.1 Wir betrachten das Alphabet $\Sigma_A = \{a, b\}$, sowie die Sprache

$L_A = \{a^n b^m \mid n, m \in \mathbb{N}_0, n > 2m\} = \{a, aa, aaa, \ldots, aaab, aaaaab, \ldots, aaaaabb, aaaaaabb, \ldots\}$.

Weiterhin betrachten wir die Grammatik $G_A = (N, \Sigma_A, P, S)$ mit $N = \{S\}$ und der Produktionsmenge

$$P: \begin{array}{ll} S & \to aaSb \\ S & \to aS \\ S & \to a \end{array}$$

(a) (2 Punkte) Begründen Sie, weshalb die Grammatik G_A die Sprache L_A erzeugt, dass also $L_A = \mathscr{L}(G_A)$ gilt.

(b) Nutzen Sie die Grammatik G_A, um für das Wort $\omega_a = aaaaabb$

α) (2 Punkte) eine Ableitung aus dem Startsymbol S

β) (2 Punkte) den entsprechenden Syntaxbaum

anzugeben.

(c) (3 Punkte) Überführen Sie die Grammatik G_A in die Chomsky-Normalform.
Es reicht, wenn Sie die **Regelmenge** P' dieser äquivalenten Grammatik $G'_A = (N, \Sigma_A, P', S)$ in CNF angeben.

(d) Nutzen Sie die Grammatik G'_A (in CNF), um für das Wort $\omega_a = aaaaabb$

α) (3 Punkte) eine Ableitung aus dem Startsymbol S

β) (3 Punkte) den entsprechenden Syntaxbaum

anzugeben.

(e) (4 Punkte) Konstruieren Sie den Kellerautomaten (oder Heuschoberautomaten) P_A (PDA oder DPDA), welcher

$$L_A = \{a^n b^m \mid n, m \in \mathbb{N}_0, n > 2m\} = \{a, aa, aaa, \ldots, aaab, aaaaab, \ldots, aaaaabb, aaaaaabb, \ldots\}$$

akzeptiert.

(f) (6 Punkte) Konstruieren Sie die Turing-Maschine T_A (TM oder NTM), welche L_A akzeptiert.

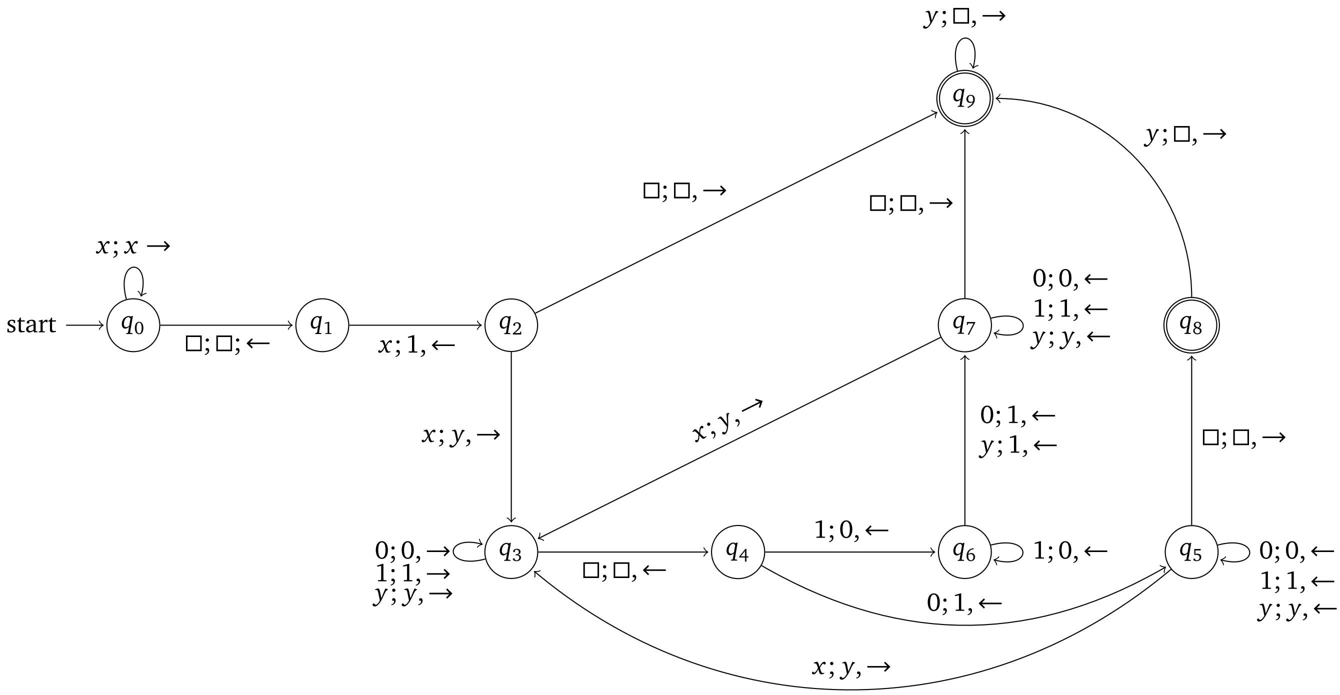


Abbildung 1: Erweitertes Zustandsübergangsdiagramm für T_x

AUFGABE 5, 24 PUNKTE
BERECHENBARKEIT, ENTSCHEIDBARKEIT & KOMPLEXITÄT

5.1 Wir betrachten das Alphabet $\Sigma_x = \{x\}$ und die Funktion $f_x : \Sigma_x^* \rightarrow \{0,1\}^*$ welche von der Turing-Maschine $T_x = (Q, \Sigma_x, \Pi_x, \delta, q_0, F) = (\{q_0, q_1, \ldots, q_9\}, \{x\}, \{x, y, 0, 1, \square\}, \{q_8, q_9\}, \delta)$ mit δ gegeben durch Abbildung 1 berechnet wird.

(a) (5 Punkte) Bestimmen Sie für die Worte $\omega_1 = \mathbf{x}$ (2 Punkte) und $\omega_2 = \mathbf{xx}$ (3 Punkte) jeweils alle Konfigurationen welche die TM T_x während der Verarbeitung der Worte durchläuft. Kürzen Sie sehr lange, uninteressante Berechnungsabschnitte durch „...“ bzw. „*“ ab!!

(b) (9 Punkte) Geben Sie für jedes der in der nebenstehenden Tabelle angegebenen Eingabewörter $\omega_i, i \in \{0, 1, \ldots, 8\}$ das von T_x berechnete Ergebnis $f_x(\omega_i)$ an.
Falls Sie der Meinung sind, dass T_x für ein Eingabewort ein undefiniertes Ergebnis liefert, verwenden Sie für das entsprechende Ergebnis das Symbol „$\perp$“.

(c) (2 Punkte) Beschreiben Sie die Funktion f_x, welche von der TM T_x berechnet wird. Konkret: was ist der Output von T_x für einen zulässigen Input?

ω_i	$f_x(\omega_i)$
$\omega_0 = \varepsilon$	
$\omega_1 = x$	
$\omega_2 = xx$	
$\omega_3 = xxx$	
$\omega_4 = xxxx$	
$\omega_5 = xxxxx$	
$\omega_6 = xxxxxx$	
$\omega_7 = xxxxxxx$	
$\omega_8 = xxxxxxxx$	

5.2 (8 Punkte) Vervollständigen Sie den folgenden Lückentext. Die Länge des Feldes sagt wenig über die Länge des einzusetzenden Textes aus. Falls Sie eine Lücke leer lassen möchten, kennzeichnen Sie dies z.B. durch „–“. **Leere Lücken geben keine Punkte.**

Turing-Maschinen akzeptieren nicht nur Sprachen, sie berechnen auch Funktionen . Allerdings kann eine Turing-Maschine nur Funktionen berechnen, ______________________________. Dies hat sich ________________________ geändert, seit die Überlegenheit der Quantencomputer (Quantum Supremacy) bewiesen wurde: Quantencomputer können ____________________________________ als herkömmliche Computer.

Unentscheidbare Probleme sind keine Erfindung. Es gibt Problem die unentscheidbar sind, da ____. Für alle entscheidbaren Probleme oder formalen Sprachen kann die Zeit- und Raumkomplexität bestimmt werden. Vor allem die Zeitkomplexität ist wichtig, da nur Probleme aus der Klasse P ________________________. Probleme aus der Klasse NP, und vor allem die NP-vollständigen Probleme sind ______________________________. Wenn Sie jedoch trotzdem eine effiziente Polyonomialzeitlösung für ein NP-vollstängiges Problem finden, ____. Ein beispielhaftes NP-vollständiges Problem ist ______________________________.