Hochschule Braunschweig/Wolfenbüttel
Institut für Angewandte Informatik
Prof. Dr. F. Seutter

Theoretische Informatik – Übungsaufgaben

1. Formale Sprachen

1.1 Sei $\Sigma = \{a, b, c\}$ ein Alphabet.

(a) Listen Sie alle Wörter über Σ mit $|w| \leq 2$ auf.

(b) Wie viele Sprachen $L \subseteq \{w \mid w \in \Sigma^*, |w| = 2\}$ gibt es? Spezifizieren Sie drei davon.

(c) Definieren Sie drei unendliche Sprachen über Σ.

(2+2+2 Punkte)

1.2 $L_1 = \{0^i \mid i \in \mathbb{N}_0\}$ und $L_2 = \{1^i \mid i \in \mathbb{N}_0\}$ seien formale Sprachen über dem Alphabet $\Sigma = \{0, 1\}$. Berechnen Sie:

(a) $L_1 \cup L_2$

(b) $L_1 \cap L_2$

(c) $L_1 \setminus L_2$

(d) $L_1 \cap \Sigma^*$

(e) $(L_1 \cup L_2) \cap \Sigma^3$

(1+1+1+1+1 Punkte)

1.3 Sei $\Sigma = \{\$, \%, \&\}$ ein Alphabet.

(a) Definieren Sie eine lineare Ordnung auf Σ.

(b) Listen Sie alle Wörter w über Σ mit $|w| \leq 2$ in lexikographischer Ordnung bzgl. der unter (a) definierten linearen Ordnung auf.

(c) Welche Wörter gehören zur Sprache $L = \{w \mid w \in \bigcup_{i=0}^{2} \Sigma^i, w = w^R\}$?

(1+2+2 Punkte)

1.4 Sei Σ ein Alphabet aus n Zeichen, $n \in \mathbb{N}$.

(a) Wie viele Wörter enthält Σ^m, $m \in \mathbb{N}_0$?

(b) Wie viele Wörter enthält $\bigcup_{i=0}^{m} \Sigma^i$, $m \in \mathbb{N}_0$?

(c) Wie viele Wörter enthält Σ^*?

(1+2+1 Punkte)

2. Endliche Automaten

2.1 Es ist ein deterministischer endlicher Automat E anzugeben, der genau diejenigen Wörter über dem Alphabet $\{0, 1, 2\}$ akzeptiert, in denen die Anzahl der Zeichen 0 und 1 jeweils gerade und die Anzahl des Zeichens 2 ungerade ist. Spezifizieren Sie E als Zustandsgraph.

(5 Punkte)

2.2 Es ist ein deterministischer endlicher Automat E zu konstruieren, der Wörter über dem Alphabet $\Sigma = \{A, B, C, \ldots, Z\}$ verarbeitet und dabei genau die erkennt, in denen *MEALY* oder *MOORE* Teilwörter sind. Spezifizieren Sie E als Zustandsgraph.

(5 Punkte)

2.3 Es sei die Sprache

$$L = \{w \mid w \in \{a, b\}^*, \text{ das vorletzte Zeichen in } w \text{ ist ein } b\}$$

gegeben.

(a) Konstruieren Sie einen nichtdeterministischen endlichen erkennenden Automaten N, der die Sprache L erkennt.

(b) Konstruieren Sie, ohne das Verfahren *Teilmengen-Konstruktion* anzuwenden, einen deterministischen endlichen erkennenden Automaten E, der die Sprache L erkennt.

Spezifizieren Sie N und E als Zustandsgraphen.

(2+3 Punkte)

2.4 Gegeben sei der nichtdeterministische endliche Automat $N = (\{1, 2, 3\}, \{a, b\}, \delta, 1, \{2\})$ mit

δ:

		a	b	ϵ
$\rightarrow$	1	$\{3\}$	$\emptyset$	$\{2\}$
$*$	2	$\{1\}$	$\emptyset$	$\emptyset$
	3	$\{2\}$	$\{2, 3\}$	$\emptyset$

Konstruieren Sie nach dem Verfahren *Teilmengen-Konstruktion* einen zu N äquivalenten deterministischen endlichen Automaten E. Spezifizieren Sie E als Zustandsgraph.

(5 Punkte)

3. Reguläre Sprachen

3.1 Stellen Sie die nachfolgenden Sprachen als reguläre Ausdrücke dar:

(a) $L = \{\, w \mid w \in \{0,1\}^* \text{ und } w \text{ beginnt mit 0 und endet mit 1} \,\}$

(b) $L = \{\, w \mid w \in \{0,1\}^* \text{ und } w \text{ enthält 11 mindestens einmal} \,\}$

(c) $L = \{\, w \mid w \in \{0,1\}^* \text{ und } w \text{ enthält 11 genau einmal} \,\}$

(d) $L = \{\, w \mid w \in \{0,1\}^* \text{ und } w \text{ enthält 11 höchstens einmal} \,\}$

(1+1+2+2 Punkte)

3.2 Stellen Sie die nachfolgenden Sprachen als deterministische endliche Automaten und als reguläre Ausdrücke dar:

(a) $L = \{\, w \mid w \in \{0,1\}^* \text{ und } w \text{ enthält das Teilwort 110} \,\}$

(b) $L = \{\, w \mid w \in \{0,1\}^* \text{ und } w \text{ enthält nicht das Teilwort 110} \,\}$

(2+2 Punkte)

3.3 Konstruieren Sie einen nichtdeterministischen endlichen Automaten, der die Sprache $L(a(abb)^*|b)$ erkennt. Verwenden Sie dazu das im Beweis zu Lemma 3.6 gegebenen Verfahren.

(5 Punkte)

3.4 Zeigen Sie mit Hilfe des Pumping Lemmas, dass die Sprache $L = \{wcw^R \mid w \in \{a,b\}^*\}$ nicht regulär ist. L ist eine Sprache über dem Alphabet $\{a,b,c\}$.

(5 Punkte)

4. Kontextfreie Sprachen

4.1 Konstruieren Sie kontextfreie Grammatiken, die genau die Sprachen

(a) $L_1 = \{a^n b^n c^m \mid m, n \in \mathbb{N}_0\}$ und

(b) $L_2 = \{a^m b^n c^n \mid m, n \in \mathbb{N}_0\}$

erzeugen.

(2+2 Punkte)

4.2 In einigen Programmiersprachen kann eine Anweisung (*statement*) auch ein Block (*block*) sein. Ein Block ist eine Anweisungsliste, die von den Zeichen { und } umgeben ist. Der Einfachheit halber sei als sonstige Anweisung nur das Terminalzeichen `stmt` mit einem abschließenden Semikolon möglich.

(a) Definieren Sie eine kontextfreie Grammatik, die diesen Ausschnitt aus einer Programmiersprache erzeugt.

(b) Ihre Grammatik sollte das folgende Wort erzeugen können:

```
{ stmt; stmt; { stmt; } }
```

Geben Sie eine Ableitung für dieses Wort an.

(4+1 Punkte)

4.3 Es sei der Kellerautomat $K = (\{1, 2, 3, 4, 5, 6, 7\}, \{a, b, c\}, \{a, \$\}, \delta, 1, \{4, 7\})$ gegeben, der die kontextfreie Sprache

$$L = \{a^i b^j c^k \mid i, j, k \geq 0 \text{ und } i = j \text{ oder } i = k\}$$

erkennt. Die Überführungsfunktion δ ist wie folgt definiert:

Q	Σ_ϵ	Γ_ϵ	$\mathcal{P}(Q \times \Gamma_\epsilon)$
1	ϵ	ϵ	$\{(2, \$)\}$
2	a	ϵ	$\{(2, a)\}$
2	ϵ	ϵ	$\{(3, \epsilon), (5, \epsilon)\}$
3	b	a	$\{(3, \epsilon)\}$
3	ϵ	$\$$	$\{(4, \epsilon)\}$
4	c	ϵ	$\{(4, \epsilon)\}$
5	b	ϵ	$\{(5, \epsilon)\}$
5	ϵ	ϵ	$\{(6, \epsilon)\}$
6	c	a	$\{(6, \epsilon)\}$
6	ϵ	$\$$	$\{(7, \epsilon)\}$

(a) Zeichnen Sie den Überführungsgraphen von K.

(b) $aabcc \in L(K)$? Falls ja, geben Sie seine Berechnung durch K an.

(c) $abbcc \in L(K)$? Falls ja, geben Sie seine Berechnung durch K an.

(1+3+1 Punkte)

4.4 Sei

$$\texttt{id = (}^n \texttt{ expr)}^n\texttt{;}$$

eine Zuweisungsanweisung einer Programmiersprache. Das zugrunde liegende Alphabet sei $\Sigma = \{\texttt{id}, \texttt{expr}, \texttt{(}, \texttt{)}, \texttt{=}, \texttt{;}\}$ und $n \in \mathbb{N}$.

Beachten Sie, dass die Anzahlen der `(` und `)` gleich sein müssen. Sei L die Sprache aller solcher Zuweisungsanweisungen.

Definieren Sie einen Kellerautomaten, der L erkennt.

(5 Punkte)

5. Turingmaschinen und Berechenbarkeit

5.1 Konstruieren Sie eine Turingmaschine T, die die Sprache $L = \{wxw \mid w \in \{a,b\}^*\}$ entscheidet. L ist eine Sprache über dem Alphabet $\{a,b,x\}$.

(a) Geben Sie zunächst eine informelle Beschreibung von T an.

(b) Definieren Sie T formal.

(c) Geben Sie den Überführungsgraphen von T an.

(d) Notieren Sie die Berechnung von T für das Wort $abxab$.

(2+4+1+1 Punkte)

5.2 Gegeben sei die Funktion $odd : \mathbb{N}_0 \to \{0,1\}$ mit

$$odd(n) = \begin{cases} 0, & \text{falls} \quad n \text{ gerade} \\ 1, & \text{falls} \quad n \text{ ungerade} \end{cases}$$

Dabei wird das Argument $n \in \mathbb{N}$ als Binärzahl ohne führende Nullen dargestellt und $n = 0$ als 0. Konstruieren Sie eine Turingmaschine T_{odd} mit $f_{T_{odd}} = odd$.

Seien Sie sich bewusst, dass die zu konstruierende Turingmaschine eigentlich die partielle Funktion $f_{T_{odd}} : \{0,1\}^* \to \{\sqcup, 0, 1\}^*$ mit

$$f_{T_{odd}}(w) = \begin{cases} odd(n), & \text{falls} \quad w = n \text{ korrekt dargestellte Binärzahl} \\ \text{undefiniert}, & \text{sonst} \end{cases}$$

berechnet.

(a) Geben Sie zunächst eine informelle Beschreibung von T_{odd} an.

(b) Definieren Sie T_{odd} formal.

(2+3 Punkte)

5.3 Spezifizieren Sie eine informelle Beschreibung einer Turingmaschine zur Berechnung der Funktion $f : \mathbb{N}_0 \times \mathbb{N}_0 \to \mathbb{N}_0$ mit $f(a,b) = a + b$. Dabei seien $a, b \in \mathbb{N}_0$ als Binärzahlen ohne führende Nullen dargestellt.

Der Einfachheit halber kann hier eine Turingmaschine mit beidseitig unendlichem Band angenommen werden. Es kann gezeigt werden, dass Turingmaschinen mit nur einseitig unendlichem Band äquivalent zu Turingmaschinen mit einer derartigen Erweiterung sind.

(5 Punkte)

5.4 Erläutern Sie, warum die folgende informelle Beschreibung einer Turingmaschine nicht effektiv und damit ungültig ist.

Informelle Beschreibung der TM T:

Eingabe: Polynom p über den Variablen $x_1, \ldots, x_n \in \mathbb{Z}$

1. Generiere alle möglichen Belegungen von $x_1, \ldots, x_n$.
2. Werte das Polynom p mit diesen Belegungen aus.
3. Falls einer dieser Werte 0 ergibt, so terminiere akzeptierend, andernfalls terminiere verwerfend.

(2 Punkte)

6. Entscheidbarkeit

6.1 Es sei der reguläre Ausdruck $R \equiv b^*(ab^*ab^*ab^*)^*$ über dem Alphabet $\{a, b\}$ gegeben.

(a) Ist $\langle R, babbaababbbaba\rangle \in L_{RA}$?
(b) Ist $\langle R, baabaababbbaba\rangle \in L_{RA}$?
(c) Ist $\langle R, babb^*aab^*\rangle \in L_{RA}$?
(d) Ist $\langle R\rangle \in L_{RA}$?
(e) Ist $\langle R, \epsilon\rangle \in L_{RA}$?

Begründen Sie Ihre Antworten.

(1+1+1+1+1 Punkte)

6.2 Betrachten Sie das Problem, ob ein deterministischer endlicher Automat E ein Wort w erkennt.

(a) Definieren Sie das Problem in Form einer Sprache L_{DEA}.
(b) Zeigen Sie, dass L_{DEA} entscheidbar ist.
(c) Modifizieren Sie das Problem, indem Sie den deterministischen durch einen nichtdeterministischen endlichen Automaten ersetzen. Entscheidet Ihr unter (b) spezifizierter Algorithmus ohne weitere Änderungen auch das modifizierte Problem? Begründen Sie Ihre Antwort.

(2+2+2 Punkte)

6.3 Es sei die unendliche Menge $\mathbb{N}^2 = \{(n, m) \mid n, m \in \mathbb{N}\}$ gegeben. Zeigen Sie, dass $\mathbb{N}^2$ abzählbar ist.

(4 Punkte)

6.4 Es sei B die unendliche Menge, die aus allen unendlichen Zeichenfolgen über $\{0, 1\}$ besteht. Zeigen Sie mittels Diagonalverfahren, dass B überabzählbar ist.

(5 Punkte)