Matrikel								SKZ				Name	

Klausur 3

Formale Grundlagen 2

25. Januar 2002

Zu jedem Buchstaben muß entweder ja oder nein angekreuzt werden.

Aufgabe 1

Betrachten Sie die folgenden Probleme.

Problem A: *Akzeptiert eine Turingmaschine das Wort* 101*?*

Problem B: *Akzeptiert eine Turingmaschine das Wort* 101 *in höchstens 1000 Schritten?*

Fragen:

A		nein

Ist **A** *entscheidbar?*

Satz von Rice (S. 67, Satz 4.2.1)

B	ja	

Ist **B** *entscheidbar?*

Eingabe von 101 *in eine Turingmaschine M, 1000 Schritte laufen lassen.*

C	ja	

Ist **A** *semientscheidbar?*

Man lasse die gegebene TM mit Input „101" laufen.

D		nein

Ist das Problem „$L(M_1) \subseteq L(M_2)$" für beliebige Turingmaschinen M_1, M_2 entscheidbar?

Satz von Rice

E		nein

Sei M_1 eine fixe Turingmaschine. Ist das Problem „$L(M_1) \subseteq L(M_2)$" für eine beliebige Turingmaschine M_2 entscheidbar?

Wähle M_1 so, daß $\emptyset \neq L(M_1) \neq \Sigma^\star$. Dann Satz von Rice.

Aufgabe 2

Gegeben sei eine Turingmaschine M mit folgender Eigenschaft:
Wenn M ein Wort akzeptiert, dann tut sie das in weniger als 1000 Schritten.

Fragen:

A	ja	

Ist $L(M)$ rekursiv?

Sei M so eine TM. Sei w irgendein Wort. Wenn, nach Eingabe von w, die Maschine in weniger als 1000 Schritten stoppt, dann hat sie entweder einen Finalzustand erreicht - $w \in L(M)$ - oder nicht - $w \notin L(M)$. Falls M nach 1000 Schritten nicht gehalten hat, muß $w \notin L(M)$ gelten, da, auf Grund ihrer speziellen Art, sie ja sonst in < 1000 Schritten gestoppt hätte. Die obige Beschreibung ist eine Konstruktionsanweisung für eine TM, welche $L(M)$ erkennt, i.e., $L(M)$ ist rekursiv.

BEMERKUNG: *Das ist keine Eigenschaft der Sprache, es ist eine Eigenschaft von Turingmaschinen. Satz von Rice hier nicht anwendbar*

B	ja	

Ist $L(M)$ rekursiv aufzählbar?

Natürlich, da rekursiv

C		nein

Ist $L(M)$ notwendigerweise endlich?

Nein. Denkbar wäre eine Turingmaschine M, die genau die Worte akzeptiert, welche mit 11 *beginnen. Sie checkt das sicher in < 1000 Schritten, aber $L(M)$ ist unendlich.*

D	ja	

Ist die Eigenschaft von $L(M)$, das leere Wort zu enthalten, entscheidbar?

Gegebene TM auf dem leeren Wort 1000 Schritte laufen lassen.

BEMERKUNG: *Dieses ist kein Problem rekursiv aufzählbarer Sprachen. Es ist eine konkrete Turingmaschine gegeben, die, auf Grund ihrer speziellen Natur, einen Entscheidungsalgorithmus zuläßt.*

Aufgabe 3

Seien a, b verschiedene positive reelle Zahlen. f sei die Polynomfunktion $f(n) = 3n^4 + 4n^2 + 5$ auf $\mathbb{N}$.

Fragen:

A	ja	

Ist $f(n)$ von der Ordnung $f(n)$?

Zu zeigen ist:

$$\exists c > 0\ \exists N \in \mathbb{N}\ \forall n \geq N\ \ (f(n) \leq cf(n))$$

Das gilt natürlich.

B	ja	

Sind $\log_a(n)$ und $\log_b(n)$ von gleicher Ordnung?

$\log_a n = \log_a b \cdot \log_b n$

C		nein

Sind a^n und b^n von gleicher Ordnung?

etwa für $a = 10,\ b = 100$, gilt $b^n = 10^n \cdot a^n$. Der Faktor 10^n übersteigt jede Konstante.

D	ja	

Gilt „$\log(n)$ ist $\mathcal{O}(f(n))$?"

$\deg\ f(n) = 4$ *und* $\mathrm{lcoeff}(f(n)) > 0$, *daher* $\lim_{x\to\infty} \frac{f(x)}{x} = \infty$.
Daher $\exists N \in \mathbb{N}\ \forall x \leq N\ \frac{f(x)}{x} \geq 1$. Für $n \geq N$ gilt somit $n \leq f(n)$, daher $n \leq 4^n \leq 4^{f(n)}$, $\log_4 n \leq f(n)$, das heißt, $\log_4$ ist $\mathcal{O}(f)$.

Aufgabe 4

Standardisierte Kodierung von Turingmaschinen:
Gegeben Turingmaschine M mit $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $Q = \{q_1, \ldots, q_n\}$. Anfangszustand q_1, Endzustandsmenge $F = \{q_2\}$, Überführungsfunktion δ. Wir bezeichnen die Symbole $0, 1, \sqcup$ in dieser Reihenfolge mit X_1, X_2, X_3, die Bewegungsrichtungen L, R mit D_1, D_2 (die stationäre Option 'S' kommt nicht vor). Eine Operation $\delta(q_i, X_j) = (q_k, X_l, D_m)$ wird kodiert als $0^i10^j10^k10^l10^m$. Die Turingmaschine M selbst wird kodiert als

$$111code_1 11code_2 \ldots 11code_r 111$$

wobei jedes $code_n$ die obige Form hat.
Der Code einer Turingmaschine M ist

$$\begin{aligned}\langle M\rangle \quad = \quad & 1110100101001001100010001000010001011000001001\\ & 001000100110101000100100110001001000100100111\end{aligned}$$

Weiters sei L_u die universelle Sprache über dem Alphabet $\{0, 1\}$.

Fragen:

A	ja	

Ist L_u rekursiv aufzählbar?

S. 65, Satz 4.1.2

B		nein

Gilt $L(M) = L_u$?

Nein. $101 \in L(M)$, *aber* $101 \notin L_u$.

C		nein

Ist L_u rekursiv?

S. 65, Satz 4.1.3

D	ja	

Gilt $11011 \in L(M)$?

Eingabe von 11011 terminiert in Finalzustand q_2.

E	ja	

Ist $\langle M\rangle 11011 \in L_u$?

Das sagt dasselbe aus wie der letzte Punkt.

F		nein

Betrachten Sie M als eine Turingmaschine, die eine Funktion berechnet. Eine natürliche Zahl n werde auf dem Band von M kodiert als eine Folge von n 1-Symbolen, das Symbol 0 diene als Trennsymbol der Eingabeparameter. Berechnet M die Funktion Funktion $f(x, y) = x + y + 2$?

δ	0	1	$\sqcup$
q_1	$(q_3, 1, R)$	$(q_1, 1, R)$	$-$
q_2	$-$	$-$	$-$
q_3	$-$	$(q_3, 1, R)$	$(q_4, \sqcup, L)$
q_4	$-$	$(q_2, \sqcup, R)$	$-$

M berechnet die Addition $(k, l) \mapsto k + l$.

G	ja	

Betrachten Sie M als eine Turingmaschine, die eine Funktion berechnet. Eine natürliche Zahl n werde auf dem Band von M kodiert als eine Folge von n 1-Symbolen, das Symbol 0 diene als Trennsymbol der Eingabeparameter. Berechnet M die Funktion $f(x, y) = x + y$?