Institut für Informatik
M. Mutz

Übungsblatt 9

Theoretische Informatik

Bearbeitungszeit: 10.06.2024 bis 16.06.2024, 16:00 Uhr
Besprechung: 17.06.2024, 10:30 Uhr in Hörsaal 5E

Abgabe: als PDF über das ILIAS
Gruppenabgaben möglich und erwünscht

Aufgabe 1 (Turingberechenbarkeit I)10P

Gegeben sei die Turingmaschine $M = (\Sigma, \Gamma, Z, \delta, z_0, \square, F)$ mit $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \square\}$, $Z = \{z_0, z_1, z_2, z_3, z_4, z_e\}$, $F = \{z_e\}$ und der untenstehenden Überführungsfunktion δ.
Die Turingmaschine berechnet eine Funktion.

δ	z_0	z_1	z_2	z_3	z_4
0	$(z_0, 0, R)$	$(z_2, 0, L)$	$(z_2, 1, L)$	$(z_3, 0, L)$	$(z_e, \square, R)$
1	$(z_0, 1, R)$	$(z_2, 1, L)$	$(z_3, 0, L)$	$(z_3, 1, L)$	$(z_e, 1, N)$
$\square$	$(z_1, \square, L)$		$(z_2, \square, N)$	$(z_4, \square, R)$	

Die passende unvollständige Zustandsbeschreibung ist in folgender Tabelle angegeben.

z_0	
z_1	
z_2	
z_3	
z_4	Gegebenenfalls eine führende Null entfernen, fertig
z_e	Endzustand

a) Füllen Sie die Zustandsbeschreibung für die Zustände z_0, z_1, z_2 und z_3 aus.

b) Die Turingmaschine M berechnet eine partielle Funktion $f : \mathbb{N} \to \mathbb{N}$ definiert durch

$$f(n) = \begin{cases} & \text{falls } \ldots, \\ \text{undefiniert} & \text{falls } \ldots. \end{cases}$$

$\mathbb{N}$ bezeichnet dabei wie im Skript die natürlichen Zahlen inklusive der 0.
Vervollständigen Sie die Funktion f.

c) Geben Sie eine vollständige Konfigurationenfolge für die Berechnung von $f(8)$ an.

d) Betrachten Sie M als Akzeptor. Geben Sie $L(M)$ formal als Menge von Wörtern an.

Lösungsvorschlag:

a) Zustandsbeschreibung:

z_0	Start, ans Wortende gehen
z_1	Ein Zeichen nach links gehen
z_2	Vorletztes Zeichen betrachten. Falls $0 \rightsquigarrow 1$ und weiter nach links; falls $1 \rightsquigarrow 0$ und in z_3. Falls es kein zweites Zeichen gibt, undefiniert, also Endlosschleife
z_3	Fertig gerechnet, an Anfang laufen
z_4	Gegebenenfalls eine fürende Null entfernen, fertig
z_e	Endzustand

b) $f : \mathbb{N} \to \mathbb{N}$ mit

$$f(n) = \begin{cases} n-2 & \text{falls } n \geq 2, \\ \text{undefiniert} & \text{falls } n < 2. \end{cases}$$

c) $8_{10} = 1000_2$
$z_0 1000 \vdash_M 1z_0 000 \vdash_M 10z_0 00 \vdash_M 100z_0 0 \vdash_M 1000z_0 \square \vdash_M 100z_1 0\square \vdash_M 10z_2 00 \vdash_M$
$1z_2 010 \vdash_M z_2 1110 \vdash_M z_3 \square 0110 \vdash_M \square z_4 0110 \vdash_M z_e 110$

d) $L(M) = \{0^m \ \mathrm{bin}(n) \mid n \geq 2, m \geq 0\} \subseteq \{0,1\}^*$

Aufgabe 2 (Turing-Berechenbarkeit II)10P

Geben Sie eine Turingmaschine an, die die Funktion $f : \mathbb{N} \to \mathbb{N}$ mit

$$f(n) = \begin{cases} 1 & \text{falls } n \bmod 5 = 0 \\ 0 & \text{sonst} \end{cases}$$

berechnet.
Sie können hierzu gerne Übungsblatt 2 als Inspiration nehmen.

Lösungsvorschlag: Eine mögliche Turingmaschine ist $M = (\{0,1\}, \{0,1,\Box\}, Z, \delta, z_0, \Box, \{z_F\})$ mit $Z = \{z_0, z_1, z_2, z_3, z_4, z_F\}$ und δ wie folgt:

δ	z_0	z_1	z_2	z_3	z_4
0	$(z_0, \Box, R)$	$(z_3, \Box, R)$	$(z_1, \Box, R)$	$(z_4, \Box, R)$	$(z_2, \Box, R)$
1	$(z_1, \Box, R)$	$(z_2, \Box, R)$	$(z_3, \Box, R)$	$(z_0, \Box, R)$	$(z_4, \Box, R)$
$\Box$	$(z_F, 1, N)$	$(z_F, 0, N)$	$(z_F, 0, N)$	$(z_F, 0, N)$	$(z_F, 0, N)$

Aufgabe 3 (LOOP-Berechenbarkeit I)10P

Sei $f : \mathbb{N} \to \mathbb{N}$ eine Funktion, die durch das folgende `LOOP`-Programm berechnet wird:

```
x0 := 1; x2 := 1;
LOOP x1 DO
    x3 := 0;
    LOOP x2 DO
        LOOP x0 DO
            x3 := x3 + 1
        END
    END;
    x0 := x3 + 0;
    x2 := x2 + 1
END
```

Zur Erinnerung: Das LOOP-Programm startet mit der Eingabe n in der Variable x_1, alle anderen Variablen sind mit 0 initialisiert und das Programm stoppt mit dem Wert $f(n)$ in der Variable x_0, wobei $n \in \mathbb{N}$.

(a) Führen Sie das LOOP-Programm für $n = 4$ aus und geben Sie $f(4)$ an. Geben Sie dabei die Werte der vier benutzten Variablen (1) vor dem ersten Durchlauf und (2) nach jedem darauf folgenden Durchlauf der äußeren LOOP-Schleife an.

(b) Beschreiben Sie zunächst informal die Bedeutung der drei LOOP-Schleifen und geben Sie dann eine formale mathematische Beschreibung für die Funktion f an.

Lösungsvorschlag:

(a) - **(3 Punkte)** Die Variablen nehmen während der Durchführung des Programms die folgenden Werte an.

	x_0	x_1	x_2	x_3
(1)	1	4	1	0
(2) Durchlauf 1	1	4	2	1
(2) Durchlauf 2	2	4	3	2
(2) Durchlauf 3	6	4	4	6
(2) Durchlauf 4	24	4	5	24

Daher ist $f(4) = 24$, nämlich der Wert von x_0 nach dem vierten Durchlauf der äußeren LOOP-Schleife.

(b) - **(2 Punkte)** Die zwei inneren ineinander geschachtelten LOOP-Programme berechnen bei jedem Durchlauf der äußeren LOOP-Schleife das Produkt von x_0 und x_2 und speichern das Ergebnis in x_3. Das LOOP-Programm kann also wie folgt umgeschrieben werden:

$x_0 := 1;\ x_2 := 1;$

LOOP x_1 DO

$\quad x_0 := x_0 * x_2;$

$\quad x_2 := x_2 + 1;$

END

Also f ist die folgende Funktion:

$$f : \mathbb{N} \to \mathbb{N},\ n \mapsto n!.$$

Aufgabe 4 (LOOP-Berechenbarkeit II)10P

Zeigen Sie die folgende Aussage aus der Vorlesung durch Angabe eines `LOOP`-Programms:

`IF` $x_1 = c$ `THEN` P `ELSE` P' `END` ist `LOOP`-berechenbar.

Verwenden Sie **nur** die elementaren Befehle, wie sie in der Definition von LOOP-Programmen aufgeführt sind.

Lösungsvorschlag:
`LOOP`-Programm:

$x_3 := 0;$

$x_4 := 1;$

$x_2 := x_1 - c;$

`LOOP` x_2 `DO` $x_3 := 1$ `END`;

`LOOP` x_3 `DO` P'; $x_4 := 0$ `END`;

`LOOP` x_4 `DO`

$x_1 := x_1 + 1;$
$x_2 := 1;$
$x_3 := 1;$
$x_1 := x_1 - c;$
`LOOP` x_1 `DO` $x_2 := 0$ `END`;
`LOOP` x_2 `DO` P'; $x_3 := 0$ `END`;
`LOOP` x_3 `DO` P `END`

`END`