Übungsblatt 5

Turing-Maschinen

{Theoretische Informatik}@AIN3

Prof. Dr. Barbara Staehle

Sommersemester 2023

HTWG Konstanz

Aufgabe 5.1 4 Punkte

Betrachten Sie die in Abbildung 1 spezifizierte Turing-Maschine $T_M = (Q, \Sigma, \Pi, \delta, q_1, F) = (\{q_1, q_2, \ldots, q_{10}\}, \{a, b, c\}, \{a,$
Ignorieren Sie bei Ihren Überlegungen den Zustand q_{10} der nie erreicht wird. Beachten Sie, dass das Symbol $\sqcup$ für ein Blank steht.

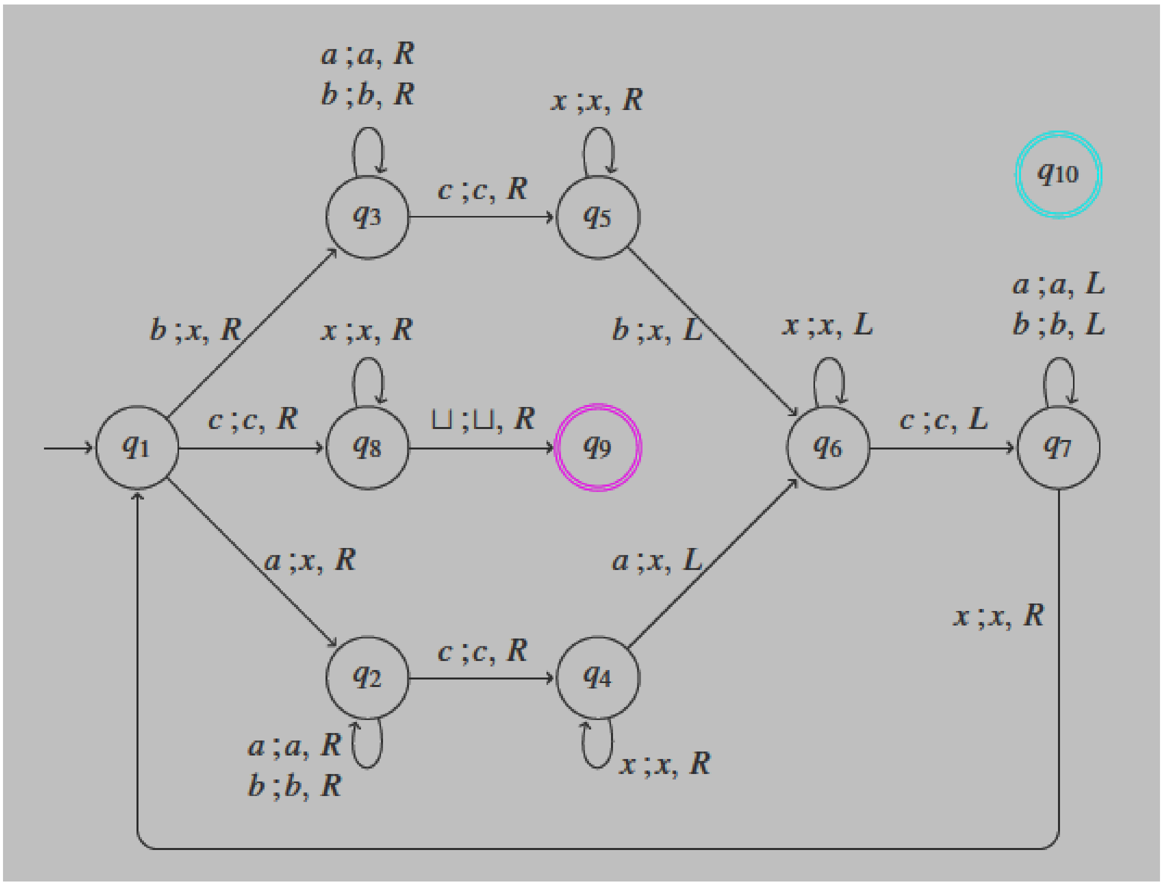

Abbildung 1: Spezifikation der Turing-Maschine T_M (Quelle: Mossakowski, Uni Magdeburg)

Bestimmen Sie für die Worte

$\omega_0 = ccc \quad \omega_1 = aca \quad \omega_2 = bbb \quad \omega_3 = abc \quad \omega_4 = abcab$

jeweils alle Konfigurationen welche die TM T_M während der Verarbeitung der Worte, ausgehend von der jeweiligen Startkonfiguration, durchläuft. Geben Sie jeweils an, ob die Worte akzeptiert werden oder nicht. Geben Sie weiterhin die von T_M akzeptierte Sprache, $\mathscr{L}(T_M)$ an.

Aufgabe 5.2 Die Sprache L_2

Wir betrachten die Sprache $L_2 = \{0^n 1^{2n} \mid n \in \mathbb{N}\} = \{011, 001111, 000111111, \ldots\}$. L_2 enthält also Strings, die aus beliebig vielen 0en gefolgt von doppelt so vielen 1en bestehen.

Teilaufgabe 5.2.1 4 Punkte

Konstruieren Sie die **Turing-Maschine (TM oder NTM)**, welche die Sprache L_2 akzeptiert. Begründen Sie es, falls das nicht möglich sein sollte.

Teilaufgabe 5.2.2 3 Punkte

Konstruieren Sie den **Kellerautomaten (PDA oder DPDA)**, welcher die Sprache L_2 akzeptiert. Begründen Sie es, falls das nicht möglich sein sollte.

Teilaufgabe 5.2.3 2 Punkte

Konstruieren Sie den **endlichen Automaten (DEA oder NEA)**, welche die Sprache L_2 akzeptiert. Begründen Sie es, falls das nicht möglich sein sollte.

Aufgabe 5.3

Gegeben Sei die Turing-Maschine $T_x = (Q, \Sigma, \Pi, \delta, q_0, F) = (\{q_0, q_1, q_a, q_b, q_f\}, \{a, b\}, \{a, b, c, \square\}, q_0, \{q_f\})$ und δ gegeben durch Tabelle 1.

δ	a	b	c	$\square$
q_0	$(q_a, c, \rightarrow)$	$(q_b, c, \rightarrow)$	-	$(q_f, c, \circlearrowleft)$
q_a	$(q_a, a, \rightarrow)$	$(q_b, a, \rightarrow)$	-	$(q_1, a, \leftarrow)$
q_b	$(q_a, b, \rightarrow)$	$(q_b, b, \rightarrow)$	-	$(q_1, b, \leftarrow)$
q_1	$(q_1, a, \leftarrow)$	$(q_1, b, \leftarrow)$	$(q_f, c, \circlearrowleft)$	-
q_f	-	-	-	-

Tabelle 1: Zustandsübergangsfunktion von T_x

Teilaufgabe 5.3.1 1 Punkt

Stellen Sie δ mit Hilfe eines erweiterten Zustandsübergangsdiagramms dar.

Teilaufgabe 5.3.2 2 Punkte

Berechnen Sie die Endkonfiguration von T_x unter der Eingabe von $\omega_1 = bb$ und $\omega_2 = aba$. Geben Sie alle Konfigurationen an, welche, ausgehend von der Startkonfiguration, durchlaufen werden.

Teilaufgabe 5.3.3 1 Punkt

a) Welche Funktion f_x berechnet T_x für ein Eingabewort $\omega \in \Sigma^*$?

b) Handelt es sich bei f_x um eine totale oder eine partielle Funktion?

c) Welche Sprache wird von T_x akzeptiert?

Begründen Sie Ihre Antworten.

AUFGABE 5.4

Wir betrachten das Alphabet $\Sigma = \{0,1\}$ und die Funktion $f_y : \Sigma^* \to \Sigma^*$ welche von der Turing-Maschine $T_y = (Q, \Sigma, \Pi, \delta, q_0, F) = (\{q_0, q_1, \ldots, q_8\}, \{0,1\}, \{0,1,\square\}, \delta, q_0, \{q_8\})$ berechnet wird.

δ sei gegeben durch:

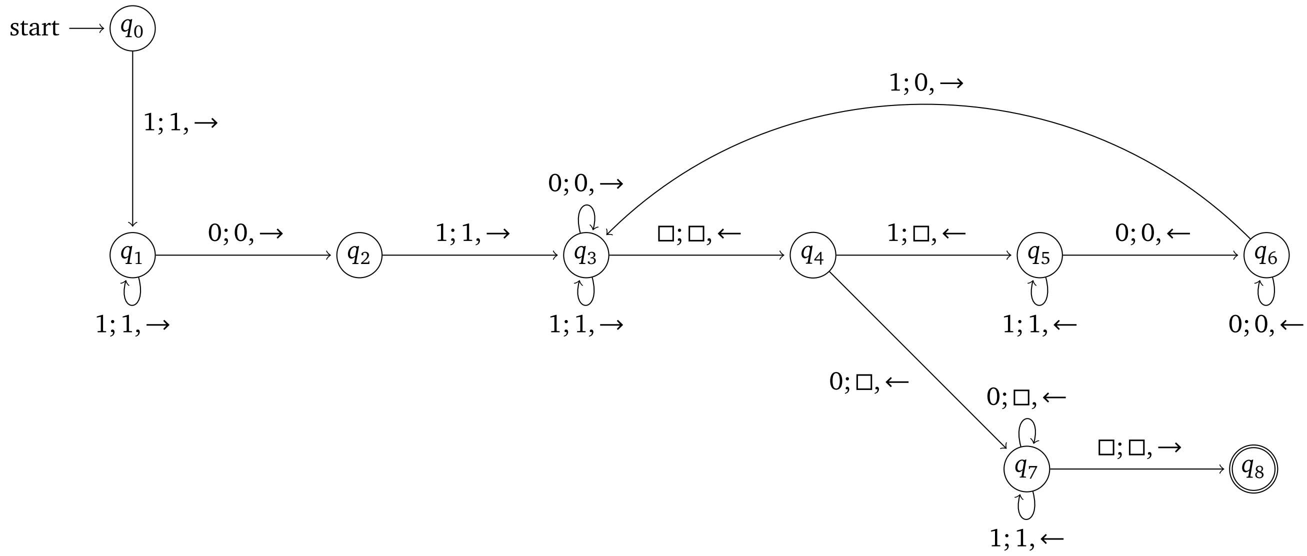

Abbildung 2: Zustandsübergangsdiagramm für T_y

TEILAUFGABE 5.4.1 3 PUNKTE

Bestimmen Sie für die Worte

$\omega_1 = 101 \quad \omega_2 = 011 \quad \omega_3 = 110 \quad \omega_4 = 1101$

jeweils alle Konfigurationen welche die TM T_y während der Verarbeitung der Worte, ausgehend von der jeweiligen Startkonfiguration, durchläuft. Geben Sie jeweils **zusätzlich** das Ergebnis der Berechnung von f_y für diese Worte an.

TEILAUFGABE 5.4.2 3 PUNKTE

Geben Sie für jedes der in der nebenstehenden Tabelle angegebenen Eingabewörter ω das von T_y berechnete Ergebnis $f_y(\omega)$ an.

Falls Sie der Meinung sind, dass T_y für ein Eingabewort ein undefiniertes Ergebnis liefert, verwenden Sie für das entsprechende Ergebnis das Symbol „$\perp$“.

Hinweis: Sie müssen **keine** durchlaufenen Konfigurationen oder sonstige Begründungen angeben!

ω	$f_y(\omega)$
ε	
0	
1	
01	
101	
11101	
11011	
111011	
111101	

Teilaufgabe 5.4.3 2 Punkte

Charakterisieren Sie die von der TM T_y berechnete Funktion f_y.

a) T_y liefert nur für manche Eingabewörter ein sinnvolles Ergebnis. Beschreiben Sie, wie ein Inputwort strukturiert sein muss, für das T_y ein Ergebnis berechnete und im Finalzustand endet.

b) Geben Sie das Ergebnis der Funktion f_y für ein gültig strukturiertes Eingabewort an, bzw. beschreiben Sie das Ergebnis für ein gültiges Eingabewort.

Teilaufgabe 5.4.4 2 Punkte

Betrachten Sie T_y als **akzeptierende** Turing-Maschine.

a) Nennen Sie (so möglich) jeweils **zwei** beispielhafte, von T_y akzeptierte, bzw. nicht akzeptierte Worte.

b) Geben Sie die von T_y akzeptierte Sprache $L_y = \mathscr{L}(T_y)$ an.

AUFGABE 5.5

Betrachten Sie die in Abbildung 3 spezifizierte Turing-Maschine A. Beachten Sie, dass das Symbol ★ für ein Blank steht.

$$A = (E, B, S, \delta, s_0, F); E = \{0, 1\}, B = \{0, 1, E, N, \star\}, S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_e\}, F = \{s_e\}$$

δ	1	0	★	E	N
s_0	(s_1, E, R)	(s_4, N, R)	$(s_e, \star, N)$		
s_1	$(s_1, 1, R)$	$(s_1, 0, R)$	(s_2, N, L)	(s_1, E, R)	(s_1, N, R)
s_2	$(s_3, 1, L)$	$(s_3, 0, L)$	$(s_5, \star, R)$	(s_2, E, L)	(s_2, N, L)
s_3	$(s_3, 1, L)$	$(s_3, 0, L)$		(s_0, E, R)	(s_0, N, R)
s_4	$(s_4, 1, R)$	$(s_4, 0, R)$	(s_2, E, L)	(s_4, E, R)	(s_4, N, R)
s_5			$(s_6, \star, L)$	$(s_5, 1, R)$	$(s_5, 0, R)$
s_6	$(s_6, 1, L)$	$(s_6, 0, L)$	$(s_e, \star, R)$		
s_e					

Abbildung 3: Spezifikation der Turing-Maschine A

TEILAUFGABE 5.5.1 3 PUNKTE

Bestimmen Sie, welches Ergebnis A unter der Eingabe von $\omega_1, \ldots \omega_3$ jeweils berechnet.

Geben Sie hierfür **zuerst** alle Konfigurationen an, welche T_x, ausgehend von der Startkonfiguration, bis zur Endkonfiguration durchläuft. Verwenden Sie hierfür die tabellarische Notation oder Konfigurationsübergänge. Geben Sie **dann** das Ergebnis der Berechnung an.

a) $\omega_1 = 1$

b) $\omega_2 = 00$

c) $\omega_3 = 01$

TEILAUFGABE 5.5.2 3 PUNKTE

a) Beschreiben Sie die Arbeitsweise von A.

b) Welche Funktion f_a berechnet A für ein Eingabewort $\omega \in E^*$? (Diese ist mathematisch schwierig zu spezifizieren, eine umgangssprachliche Beschreibung reicht).

c) Handelt es sich bei f_a um eine totale oder eine partielle Funktion?

d) Geben Sie die von A akzeptierte Sprache $\mathscr{L}(A)$ an.

AUFGABE 5.6

TEILAUFGABE 5.6.1 3 PUNKTE

Erstellen Sie die Turingmaschine T_{-3} welche die Funktion $f : \mathbb{N}_0 \to \mathbb{N}_0, n \mapsto \begin{cases} n-3 & \text{falls } n \geq 3 \\ 0 & \text{falls } n < 3 \end{cases}$ für **unär codierte Zahlen** $(0 = \varepsilon, 1 = 1, 2 = 11, 3 = 111, \ldots)$ berechnet.

Geben Sie die Zustandsübergangsfunktion δ sowohl in tabellarischer Form, als auch in Form eines erweiterten Zustandsübergangsdiagrammes an.

Achten Sie darauf, dass T_{-3}

- für die unär codierte Eingabe n die unär codierte Ausgabe $f(n)$ ausgibt und erfolgreich terminiert,
- auch $f(0), f(1), f(2), f(3)$ korrekt berechnet werden.

TEILAUFGABE 5.6.2 ARBEITSWEISE VON T_{-3}, 2 PUNKTE

Berechnen Sie die Endkonfiguration für T_{-3} unter der Eingabe von

a) $\omega = \varepsilon$

b) $\omega = 11$

c) $\omega = 1111$

Geben Sie alle Konfigurationen an, welche, ausgehend von der Startkonfiguration, durchlaufen werden.

Aufgabe 5.7 Plus Eins

Ziel dieser Aufgabe ist es, Turing-Maschine T_{b+1} und T_{d+1} zu konstruieren, welche zu einer *Binärformat* bzw. *Dezimalformat* gegeben Zahl eine 1 addieren.

Teilaufgabe 5.7.1 4 Punkte

Konstruieren Sie die TM T_{b+1}, welche unter Verwendung des Eingabealphabets $\Sigma = \{0, 1\}$ die Funktion $f_b : \mathbb{N}_0 \to \mathbb{N}_0, f(n) = n + 1$, also für eine **binär** codierte Eingabe n die **binär** codierte Ausgabe $n+1$ berechnet.

Teilaufgabe 5.7.2 5 Punkte

Verallgemeinern Sie die binär arbeitende TM T_{b+1} und konstruieren Sie die TM T_{d+1} welche unter Verwendung des Eingabealphabets $\Sigma = \{0, 1, 2 \ldots, 9\}$ die Funktion $f_d : \mathbb{N}_0 \to \mathbb{N}_0, f(n) = n + 1$, also für eine **dezimal** codierte Eingabe n die **dezimal** codierte Ausgabe $n + 1$ berechnet.

Aufgabe 5.8 Berechnung des Paritätsbits, 5 Punkte

Wir betrachten das Alphabet $\Sigma = \{0, 1\}$ und die Funktion $f_p : \Sigma^* \to \Sigma^*$ welche von der Turing-Maschine T_p berechnet wird. T_p hängt an ein nichtleeres binäres Eingabewort ein **Paritätsbit** an, verlängert einen 0/1-String also um genau ein Zeichen, um sicherzustellen, dass die Summe der 1en eine gerade Zahl ist. Konkret:

$$f_p : \Sigma^* \to \Sigma^*, \quad f_p(\omega) = \begin{cases} \omega 0 & \text{falls } \omega \text{ eine gerade Anzahl von 1en enthält,} \\ \omega 1 & \text{falls } \omega \text{ eine ungerade Anzahl von 1en enthält} \\ \perp & \text{falls } \omega = \varepsilon, \end{cases}$$

Beispiele: $f_p(0) = 00, f_p(1) = 10, f_p(101) = 1010, f_p(100) = 1001$

Konstruieren Sie die Turing-Maschine T_p welche wie spezifiziert arbeitet!