

Theoretische Informatik (WS 2005/2006) - Übungsblatt 7

Jan Hendrik Dithmar - Matrikelnummer 2031259
Do, 11-13 Uhr, Vasileios Anastasatos

Aufgabe 1

Spezifikation der Maschine ${}_{ab}T_{ccd}$:

$$\Sigma = \{a, b, c, d\}$$

$$\Gamma = \{a, b, c, d, \#\}$$

$$Q = \{A, B, C, D, W_a, W_b, q, q', W\}$$

$$s = q$$

$$F = \{q'\}$$

$$\Delta = \{(q, k, q, k, R), \quad \text{mit } k \in \{b, c, d\}$$

$$(q, a, W_a, a, R),$$

$$(W_a, b, W_b, d, L),$$

$$(W_a, k', q, k', R), \quad \text{mit } k' \in \{a, c, d\}$$

$$(W_b, a, W, c, L),$$

$$(W, c, C, c, L),$$

$$(W, a, A, c, L),$$

$$(W, d, D, c, L),$$

$$(W, b, B, c, L),$$

$$(X_1, x_1, X_2, x_2, L), \quad \text{wobei } (X_1, x_1); (X_2, x_2) \in \{(A, a); (B, b); (C, c); (D, d)\}$$

$$(X, \#, q', x, B) \quad \text{wobei } (X, x) \in \{(A, a); (B, b); (C, c); (D, d)\}$$

$$\}$$

Wirkungsweise:

Die Maschine ${}_{ab}T_{ccd}$ beginnt am linken Ende des Bandes. Von hier aus macht der Lesekopf solange „Rechtsbewegungen“ bis er auf ein a stößt. In diesem Fall wechselt die Maschine in einen Zustand W_a und prüft dann, ob der Buchstabe rechts von a ein b ist. Ist dies nicht der Fall, so wechselt sie wieder in den Startzustand q zurück und setzt das Einlesen mit „Rechtsbewegung“ fort. Tritt bis zum rechten Ende des Bandes kein ab auf, so wird die Maschine im Zustand q dort stecken bleiben.

Findet die Maschine rechts von a jedoch ein b , so ersetzt sie dieses durch ein d und geht über in eine „Linksbewegung“ und einen Zustand W_b . Jetzt wird das gefundene a durch ein c ersetzt, in Zustand W gewechselt und der Kopf weiter nach links bewegt. In Zustand W wird der aktuelle Buchstabe eingelesen und mit Hilfe besonderer Zustände (A, B, C, D) „zwischengespeichert“. In jedem Fall wird hier ein weiteres c geschrieben, so dass jetzt die Buchstabenfolge ccd komplett ist. Im Folgenden liest die Maschine in „Linksbewegung“ immer einen Buchstaben ein, speichert diesen mittels A, B, C, D kurzzeitig und schreibt an die Stelle den jeweils vorher zwischengespeicherten Buchstaben. Erreicht sie das linke Ende

so schreibt sie den letzten zwischengespeicherten Buchstaben statt #, lässt den Kopf dort stehen und wechselt in den Zustand q' .

Aufgabe 2

(a)

2NKA $M = (\Sigma, \Gamma_a, \Gamma_b, \epsilon_a, \epsilon_b, Q, s, F, \Delta)$.

Σ Eingabealphabet

Γ_a Kelleralphabet des ersten Kellers

Γ_b Kelleralphabet des zweiten Kellers

$\epsilon_a \in \Gamma_a$ Kellerbodensymbol des ersten Kellers

$\epsilon_b \in \Gamma_b$ Kellerbodensymbol des zweiten Kellers

Q endliche Zustandsmenge

$s \in Q$ Startzustand

$F \subset Q$ Endzustände

$\Delta \subset (Q \times \Gamma_a \times \Gamma_b \times (\Sigma \cup \{\varepsilon\}) \times \Gamma_a^* \times \Gamma_b^* \times Q)$

Dabei ist Q der alte Zustand, Γ_a und Γ_b das oberste Kellersymbol des jeweiligen Kellers, $(\Sigma \cup \{\varepsilon\})$ das nächste Zeichen auf dem Eingabeband, Γ_a^* und Γ_b^* der neue Kellergipfel des jeweiligen Kellers und Q der neue Zustand.

Konfigurationsraum:

$$K_M = Q \times \Gamma_a^* \times \Gamma_b^* \times \Sigma^*$$

Dabei ist Q der derzeitige Zustand, Γ_a^* und Γ_b^* der Kellerinhalt und Σ^* der Eingaberest.

Startkonfiguration: $start_x = (s, \epsilon_a, \epsilon_b, x)$ mit $x \in \Sigma^*$.

Endkonfiguration: $Fin = \{(f, U, V, \varepsilon) | f \in F, U \in \Gamma_a^*, V \in \Gamma_b^*\}$

Rechenschrittrelation:

Seien $q, q' \in Q$; $A \in \Gamma_a$; $B \in \Gamma_b$; $X, U \in \Gamma_a^*$; $Y, V \in \Gamma_b^*$; $a \in \Sigma$; $u \in \Gamma^*$.

$$(q, AU, BV, au) \vdash_M (q', XU, YV, u) \Leftrightarrow (q, A, B, a, X, V, q') \in \Delta$$

$$(q, AU, BV, u) \vdash_M (q', XU, YV, u) \Leftrightarrow (q, A, B, \varepsilon, X, V, q') \in \Delta$$

$\vdash_M^*$ ist die reflexive, transitive Hülle von $\vdash_M$.

Akzeptanz:

$$L(M) = \{x \in \Sigma^* | start_x \vdash_M^* \varphi \text{ für irgendein } \varphi \in Fin\}$$

2NKA-Sprache:

L ist genau dann 2NKA-Sprache, wenn $\exists$ 2NKA M mit $L = L(M)$.

(b)

Sei $L = L_{abc} = \{a^n b^n c^n | n \in \mathbb{N}\}$. L ist laut Vorlesung nicht NKA-Sprache.

Behauptung: L ist 2NKA-Sprache

Beweis durch Automaten:

Sei 2NKA $M = (\Sigma, \Gamma_a, \Gamma_b, \epsilon_a, \epsilon_b, Q, s, F, \Delta)$.

$$\begin{aligned}\Sigma &= \{a, b, c\} \\ \Gamma_a &= \{A\} \\ \Gamma_b &= \{B\} \\ Q &= \{s, f\} \\ F &= \{f\} \\ \Delta &= \{(s, \epsilon_a, \epsilon_b, \varepsilon, \epsilon_a, \epsilon_b, f), \\ &\quad (s, \epsilon_a, \epsilon_b, a, A\epsilon_a, \epsilon_b, s), \\ &\quad (s, A, \epsilon_a, b, \varepsilon, B\epsilon_b, s), \\ &\quad (s, \epsilon_a, B, c, \epsilon_a, \varepsilon, s), \\ &\quad (s, A, \epsilon_b, a, AA, \epsilon_b, s), \\ &\quad (s, A, B, b, \varepsilon, BB, s) \\ &\quad \}\end{aligned}$$

(c)

Behauptung: 2NKA-Sprachen gehören zum Typ0.

Beweis:

Wir wissen, dass einfache TM Typ0-Sprachen akzeptieren. Es genügt hier also zu zeigen, dass 2NKA M die selbe Sprache akzeptiert wie eine einfache TM.

Idee: Baue 2NKA, der eine einfache TM simuliert.

Auf dem 2NKA-Eingabeband stehe das selbe wie auf dem eTM-Band. Im ersten Keller des 2NKA stehe immer der Inhalt links vom Lesekopf der eTM, im zweiten Keller immer der Inhalt rechts vom Lesekopf der eTM. Diese Invarianz soll immer aufrecht erhalten bleiben.

So findet man im ersten Keller immer die Symbole bei „Linksbewegung“, analog im zweiten Keller die Symbole der „Rechtsbewegung“. Man hat also die selben Möglichkeiten, die die einfache TM bietet.