Grundlagen der Programmierung

SS 05

Prof. Dr. K. Madlener

Lösungshinweise zu Übungsblatt 10

Aufgabe 10.1. Sei $\Sigma = \{a, b\}$, $A = \{a^n b^n : n \in \mathbb{N}\}$.

a) Sei $T = (Q, \Sigma, \Gamma, \delta, q_0, F)$ gegeben durch $\Gamma = \Sigma \cup \{\square\}$, $F = \{q_y, q_n\}$
Sowie durch folgendes Programm(plus Kommentare):

$q_0 :$ R

$\cdots \underbrace{\square}_{\uparrow} cw\square$ oder $\underbrace{\square}_{\uparrow} \square$

$\cdots \square \underbrace{c}_{\uparrow} w\square$ oder $\square \underbrace{\square}_{\uparrow}$

$\square : q_y$

$\cdots \square \underbrace{c}_{\uparrow} w\square$

$b : q_n$

$\cdots \square \underbrace{a}_{\uparrow} w\square$

$\square$

$\cdots \square \underbrace{\square}_{\uparrow} w\square$

$SR\square$

$\cdots \square w \underbrace{\square}_{\uparrow}$

L

$\cdots \square w' \underbrace{d}_{\uparrow} \square$ oder $\underbrace{\square}_{\uparrow} \square$

$\square : q_n$

$\cdots \square w' \underbrace{d}_{\uparrow} \square$

$a : q_n$

$\cdots \square w' \underbrace{b}_{\uparrow} \square$

$\square$

$\cdots \square w' \underbrace{\square}_{\uparrow} \square$

$SL\square$

$\cdots \underbrace{\square}_{\uparrow} w'\square$

q_0

Beh.: Ist T zum i-ten Mal bei q_0, so war die Startkonfiguration $\underbrace{\square}_{\uparrow} a^i \hat{w} b^i \square$ bei aktueller Konfiguration $\underbrace{\square}_{\uparrow} \hat{w}\square$ mit $\hat{w} \in \sum^*$.

Bew.$i = 0$: Klar.

$i \to i+1$: Ist beim i-ten Erreichen von q_0 die Konfiguration $\underbrace{\square}_{\uparrow} \hat{w}\square$, so war (nach I.V.) $\underbrace{\square}_{\uparrow} a^i \hat{w} b^i \square$ die Startkonfiguration.

Damit q_0 zum $(i+1)$-ten Mal erreicht wird, muss $\hat{w} = a\hat{\hat{w}}b$ sein (s. Kommentare zum Programm); die Konfiguration beim $(i+1)$-ten Erreichen von q_0 ist dann (s.Kommentare) $\underbrace{\square}_{\uparrow} \hat{\hat{w}}\square$. Die Startkonfiguration war

$\underbrace{\square}_{\uparrow} a^i \hat{w} b^i \square = \underbrace{\square}_{\uparrow} a^i a\hat{\hat{w}} b b^i \square = \underbrace{\square}_{\uparrow} a^{i+1} w b^{i+1} \square$.

Stoppt T mit Zustand q_y, so ist die aktuelle Konfiguration $\square \underbrace{\square}_{\uparrow}$. Dies erfolgt, nachdem zum k-ten Male q_0 erreicht wurde (für ein $k \in \mathbb{N}$, d.h. die Startkonfiguration war $\underbrace{\square}_{\uparrow} a^k b^k \square$. Weiter ist klar: $a^k b^k \in \mathbb{A}$.

Stoppt T mit Zustand q_n nach dem k-ten Erreichen von q_0 (und ohne einem $(k+1)$-ten Erreichen), so ist die aktuelle Konfiguration $\square \underbrace{b}_{\uparrow} w \square$ (d.h. Startkonfiguration $\underbrace{\square}_{\uparrow} a^k b w b^k$) oder $\underbrace{\square}_{\uparrow} \square$(d.h. Startkonfiguration $\underbrace{\square}_{\uparrow} a^{k+1} b^k \square$) oder $\square w' \underbrace{a}_{\uparrow} \square$ (d.h. Startkonfiguration $\underbrace{\square}_{\uparrow} a^{k+1} w' a b^k \square$).

Aber $a^k b w b^k$, $a^{k+1} b^k$ und $a^{k+1} w' a b^k$ sind für kein $k \in \mathbb{N}$, $w, w' \in \sum^I$ in $\mathbb{N}$.
Weiter gilt für $x \notin \mathbb{A}$, dass entweder $x = a^r b^s$ mit $r \neq s$ oder $x = x' b a x''$ $(x', x'' \in \sum^*)$. Dies ist jeweils einer der drei obigen Fäll, in denen T mit q_n hält.
Da T in jedem Fall hält (wenn an Anfag von q_0 gilt Länge$(cw) = k+1$, gilt am Ende —vor dem letzten Befehl— Länge$(w') = k-1$ oder der neue Zustand is q_y oder q_n), gilt: T entscheidet A.

b) Wir nehmen an, dass T' eine Turingmaschine ist, die A entscheidet und nur nach rechts geht (nur nach links ist ohnehin Unsinn).

T' hält also ausgehend von Startkonfiguration $q_0 \square w \square$ in einer Konfiguration $\square w' \square^k q_? \square$ (mit $w, w' in \Sigma^*, k \in \mathbb{N}, q_? \in \{q_y, q_n\}$), da T' für $w \in A$ ganz w lesen muss.

Sei nun $w = a^n b^n$ für ein $n > |Q'|$ (Q' sei die Menge der Zustände von T'). w stehe auf den Bandfeldern 1 bis $2n$. Für $i = 1, \cdots, n$, Sei $q'_n \in Q'$ der Zustand, in dem sich T' befindet, wenn es Feld i verlässt (nach rechts, in Richtung Feld $i+1$).

Da $n > |Q'|$, gibt es $1 \leq i < j \leq n$ mit $q_i = q_j$. Wird nun der Teil des Bandes bestehend aus den Feldern $i+1$ bis j "herausgeschnitten", so geht T' im Zustand $q_i = q_j$ von Feld i auf Feld $j+1$ über; sonst ändert sich das Verhalten von T' nicht. Da T' $a^n b^n$ akzeptiert, akzeptiert T' also auch $a^{n-j+i} b^n \notin A$.
Widerspruch!!!

Aufgabe 10.2. $\Sigma = \{a, b\}$
Sei $T = (Q, \Sigma, \Gamma, \delta, q_0, F)$ gegeben durch
$\Gamma = \Sigma \cup \{\Box, \triangledown\}$
$F = \{q_f\}$
und

$q_0 :$ $SR\Box$ $\cdots \Box w \underbrace{\Box}_{\uparrow}$
$\triangledown$
R
$\triangledown$
L
$q_1 :$ $\cdots$ hier gilt $\Box w_0 \underbrace{\triangledown}_{\uparrow} w_1 \triangledown \rho(w_1), w = w_0 w_1$
L
$\Box : q_2$ $\cdots$ also ist $w_0 = \varepsilon \uparrow$.
$a : q_a$
$b : q_b$
$q_a :$ $\triangledown$
R
a
$SR\Box$
a
$SL\triangledown$
$SL\triangledown$
q_1 $\cdots \Box w_0' \underbrace{\triangledown}_{\uparrow} a w_1 \triangledown \rho(a w_1), w_0 = a w_0'$

$q_b:$ $\bigtriangledown$
R
b
$SR\square$
b
$SL\bigtriangledown$
$SL\bigtriangledown$
q_1 $\cdots \square w_0' \underbrace{\bigtriangledown}_{\uparrow} bw_1 \bigtriangledown \rho(bw_1), w_0 = bw_0'$

$q_2:$ $SR\bigtriangledown$
$\square$
$SR\bigtriangledown$
$\square$
$SL\square$
q_f $\cdots$ entferne $\bigtriangledown, AF$ auf richtige Position.

Sei im Folgenden $v \in \{a, b\}$. Sei das Eingabewort $w = v_1 \cdots v_n$.
Beh.: Ist $i \leq n$ und T zum i-ten Mal i-Zustand q_1, ausgehend von Startkonfiguration $q_0\square v_1 \cdots v_n$, mit Konfiguration k_i, dann ist

$$k_i \sim v_1 \cdots v_{n-i+1} q_1 \bigtriangledown v_{n-i+2} \cdots v_n \bigtriangledown \rho(v_{n-i+2} \cdots v_n)$$

Bew.:
$i = 1 : k_1 \sim v_1 \cdots v_n q_1 \bigtriangledown \bigtriangledown$

$i \to i+1, i < n :$ Man berechne ausgehend von k_i das Programm.
Springt man über $\square : q_2$ nach q_2 nach dem $(i+1)$-ten Mal in Zustand q_1, ist die Funktion im wesentlichen berechnet, d.h. das gespiegelte Wort steht noch, gekennzeichnet durch $\bigtriangledown$ von der Eingabe, auf dem Band.
Sonst wird in Abhängigkeit von v_{n-i+1} nach q_a oder q_b gesprungen. Die nachfolgenden Schritte entsprechen sich, d.h. man schiebt $\bigtriangledown$ um einen Buchstaben weiter nach links, sucht rechts $\square$, schreibt v_i, geht links zum ersten $\bigtriangledown$, das auf dem Band steht und springt nach q_1, d.h. es wird genau der folgende Zustand erreicht, der k_{i+1} entspricht $(\rho(v_{n-i+2} \cdots v_n) v_{n-i+1} = \rho(v_{n-i+1} \cdots v_n))$:

$$k_{i+1} \sim v_i \cdots a_{n-i} q_1 \bigtriangledown v_{n-i+1} \cdots v_n \bigtriangledown \rho(v_{n-i+1} \cdots v_n)$$

Mit dieser Beh. lässt sich nun folgern, dass T für jede Startkonfiguration $k_0 = q_0\square w$, $w \in \Sigma^*$ hält und zwar in Endkonfiguration $q_f\square w \square \rho(w)$.

Aufgabe 10.3. Sei (Σ, Π) ein WES mit $\Sigma = \{a, b\}$ und $\Pi = \{ab ::= bbb, bbb ::= ab, bb ::= b, b ::= bb\}$.

Beh.: $R \subseteq \Sigma^* \times \Sigma^*$ mit Rxy gdw $\exists n \in \mathbb{N}$ mit $x \vdash_\pi^n y$ für $x, y \in \Sigma^*$ ist entscheidbar.

Bew.: Wir zeigen zunächst: Für alle $x, y \in \Sigma^*$ gilt:
Es gibt $n \in \mathbb{N}$ mit $x \vdash_\pi^n y$ gdw. $x = y$ oder es gibt $k \in \mathbb{N}$ und $w_1, w_2 \in \Sigma^*$ mit $x = w_1ba^k$ und $y = w_2ba^k$.

$\rightarrow$: Sei $x, y \in \Sigma^*$,$n \in \mathbb{N}$ mit $x \vdash_\pi^n y$. Wir zeigen die Behauptung durch Induktion über n.

$n = 0$: $x = y$.

$n \rightarrow n + 1$: Es gibt also $z \in \Sigma^*$ mit $x \vdash_\pi^n z \vdash_\pi y$.
Nach I.V.: $z = x$ oder es gibt $k \in \mathbb{N}$, $w_1, w_2 \in \Sigma^*$ mit $x = w_1ba^k$, $y = w_2ba^k$.

(1) $z = x \rightarrow x \vdash_\pi y$.
(a)$x = uabv, y = ubbbv\, (u, v \in \Sigma^*)$. Klar: $x \neq y$.
Ist $v = a^l (l \in \mathbb{N})$, so ist $x = w_1'ba^l, y = w_2'ba^l$ mit $w_1' = ua, w_2' = ubb$.
Ist $v = wba^l (w \in \Sigma^*, l \in \mathbb{N})$, so ist $x = w_1'ba^l, y = w_2'ba^l$ mit $w_1' = uabw, w_2' = ubbbw$.
(b)$x = ubbbv$, $y = uabv\, (u, v \in \Sigma^*)$. Ähnlich.
(c)$x = ubbv$, $y = ubv\, (u, v \in \Sigma^*)$. Ähnlich.
(d)$x = ubv$, $y = ubbv\, (u, v \in \Sigma^*)$. Ähnlich.

(2) Es gibt $k \in \mathbb{N}$, $w_1, w_2 \in \Sigma^*$ mit $x = w_1ba^k$, $z = w_2ba^k$.
Wegen $z \vdash_\pi y$ gibt es $u, v \in \Sigma^*$ und $l ::= r \in \Pi$ mit $z = ulv$, $y = urv$.
(a)Sei $w_2 = ulw_2'\, (w_2' \in \Sigma^*)$, d.h. $v = w_2'ba^k$. $\rightarrow$ $y = urw_2'ba^k = w_2''ba^k$ mit $w_2'' = urw_2'$.
(b)Sei $w_2b = ul$ $\rightarrow$ $y = ura^k = w_2'ba^k\, (w_2' \in \Sigma^*)$. ur muss $w_2'b$ sein weil die rechte Seite von jedem Regel mit b endet. (c) Keine andere Möglichkeit weil die linke Seite von jedem Regel mit b endet.

(a) und (b) gelten jeweils für alle anwendbaren Produktionen.

$\leftarrow$: Sei $x, y \in \Sigma^*$ mit $x = y$ oder es gibt $k \in \mathbb{N}$, $w_1, w_2 \in \Sigma^*$, so dass $x = w_1ba^k$, $y = w_2ba^k$.
Wir zeigen die Behauptung:
Ist $x = y$ wähle $n = 0$.
Ansonsten gebe es $k \in \mathbb{N}$, $w_1, w_2 \in \Sigma^*$ mit $x = w_1ba^k$, $y = w_2ba^k$.
Wir zeigen: Für alle $k \in \mathbb{N}, w \in \Sigma^*$ gibt es $l \in \mathbb{N}$ mit $wba^k \vdash_\pi^l ba^k$ und $ba^k \vdash_\pi^l wba^k$.
(Induktion über w.)

$$
\begin{array}{ll}
w = \varepsilon: & l = 0. \\
w = w'a: & w'aba^k \vdash_\pi w'bbba^k \vdash_\pi w'bba^k \vdash_\pi w'ba^k \underbrace{\overset{l'}{\vdash}}_{\pi \ \text{I.V.}} ba^k \ (l' \in \mathbb{N}) \\
 & ba^k \underbrace{\overset{l'}{\vdash}}_{\pi \ \text{I.V.}} w'ba^k \vdash_\pi w'bba^k \vdash_\pi w'bbba^k \vdash_\pi w'aba^k \ (l' \in \mathbb{N}). \quad \text{Also gibt es} \\
 & \text{Wähle also } l = l' + 3. \\
w = w'b: & \text{Ähnlich.}
\end{array}
$$

auch $l_1, l_2 \in \mathbb{N}$ mit $w_1ba^k \vdash_\pi^{l_1} ba^k \vdash_\pi^{l_2} w_2ba^k$ oder $x \vdash_\pi^n y$ mit $n = l_1 + l_2$.
Behauptung ist sicherlich entscheidbar.

Aufgabe 10.4. Sei $g : (\Sigma^*)^{m+1} \to \Sigma^*$, $h_a : (\Sigma^*)^{m+2} \to \Sigma^*$ (f.a. $a \in \Sigma$) und es gelte $g, h_a \in \mathcal{P}(\Sigma^*)$.

Beh.: $f_r : (\Sigma^*)^{m+2} \to \Sigma^*$ mit

$$
\begin{array}{rcl}
f_r(\overrightarrow{u}, \varepsilon) & = & g(\overrightarrow{u}, \varepsilon) \\
f_r(\overrightarrow{u}, va) & = & h_a(\overrightarrow{u}, f_r(\overrightarrow{u}, v), v)
\end{array}
$$

f.a. $\overrightarrow{u} \in (\Sigma^*)^m, a \in \Sigma, v \in \Sigma^*$
ist in $\mathcal{P}(\Sigma^*)$.

Bew.: Sei $h'_a : \Sigma^{*m+2} \to \Sigma^*$ (für alle $a \in \Sigma$) gegeben durch

$$h'_a(\overrightarrow{u}, z, w) = h_a(\overrightarrow{u}, z, \rho(w)) \text{ f.a. } \overrightarrow{u} \in (\Sigma^*)^m\ z, w \in \Sigma^*.$$

(ρ ist die Spiegelungsfunktion).
Da h_a und ρ in $\mathcal{P}(\Sigma^*)$ liegen, gibt dies auch für h'_a (f.a. $a \in \Sigma$).

Beh.: Entsteht $f' : (\Sigma^*)^{m+2} \to \Sigma^*$ aus g sowie h'_a (für alle $a \in \Sigma$) durch primitive Rekursion, so gilt

$$f_r(\overrightarrow{u}, v) = f'(\overrightarrow{u}, \rho(v)) \text{ f.a. } \overrightarrow{u} \in (\Sigma^*)^m, v \in \Sigma^*.$$

Bew.: Induktion über v:

$$
\begin{array}{lll}
v = \varepsilon: & f_r(\overrightarrow{u}, \varepsilon) & = g(\overrightarrow{u}, \varepsilon) = f'(\overrightarrow{u}, \varepsilon) = f'(\overrightarrow{u}, \rho(\varepsilon)). \\
v = v'a: & f_r(\overrightarrow{u}, v'a) & = h'_a(\overrightarrow{u}, f_r(\overrightarrow{u}, v'), v') \\
 & & \underbrace{=}_{\text{I.V.}} h'_a(\overrightarrow{u}, f'(\overrightarrow{u}, \rho(v')), v') \\
 & & = h_a(\overrightarrow{u}, f'(\overrightarrow{u}, \rho(v')), \rho(v')) \\
 & & = f'(\overrightarrow{u}, a\rho(v')) \\
 & & = f'(\overrightarrow{u}, \rho(v'a))
\end{array}
$$

Da aber $f' \in \mathcal{P}(\Sigma^*)$, ist auch $f \in \mathcal{P}(\Sigma^*)$.

Informationen zur Vorlesung:
http://www-madlener.informatik.uni-kl.de/ag-madlener/teaching/ss2005/gdp/gdp.html