

Theoretische Informatik

Lösung für das Übungsblatt 2 (2025W)

Aufgabe 2.1

- Geben Sie eine Turingmaschine M (gemäß Foliensatz 2, Folie 13) an, die einen (nicht-leeren) Binärstring als Eingabe erhält und die am weitesten rechts liegende Binärziffer der Eingabe dualisiert, also 1 durch 0 ersetzt und 0 durch 1.
- Führen Sie eine Beispielrechnung für M mit dem Eingabestring 110, also $[6]_2$, durch.
Genauer: Geben Sie eine Folge von Konfigurationen an, die mit der Anfangskonfiguration q_0110 beginnt und einer Endkonfiguration endet. Wie lautet die Ausgabe als natürliche Zahl?
- Beschreiben Sie die Arbeitsweise von M für beliebige (binäre) Eingabestrings informell.
- Welche Funktion f_M vom Typ $\mathbb{N} \rightarrow \mathbb{N}$ wird von M berechnet?

Lösung 2.1

- $M = (\{q_0, q_1, q_2, q_3\}, \{0, 1\}, \{0, 1, B\}, \delta, q_0, B, \{q_3\})$, wobei δ wie folgt definiert ist:

δ	0	1	B
q_0	$(q_0, 0, R)$	$(q_0, 1, R)$	(q_1, B, L)
q_1	$(q_2, 1, L)$	$(q_2, 0, L)$	
q_2	$(q_2, 0, L)$	$(q_2, 1, L)$	(q_3, B, R)

Beachte: Die Ausgabewerte für (q_1, B) und sämtliche Ausgabewerte für q_3 sind irrelevant und wurden daher in der Tabelle für δ nicht spezifiziert.

- Die Konfigurationsfolge (Rechnung) lautet

$$\begin{aligned} q_0110 &\Rightarrow 1q_010 &\Rightarrow 11q_00 &\Rightarrow 110q_0[B] &\Rightarrow 11q_10 \\ &\Rightarrow 1q_211 &\Rightarrow [B]q_2111 &\Rightarrow q_2B111 &\Rightarrow [B]q_3111 \end{aligned}$$

Da q_3 ein Endzustand ist, hält M mit der Ausgabe 111. Es gilt $[111]_2 = 7$.

- Laut Definition steht der Lese/Schreibkopf im Anfangszustand q_0 über der ersten Binärziffer des Eingabestrings. In diesem Zustand geht M , ohne den Bandinhalt zu verändern, so weit nach rechts, bis ein Blank-Symbol (B) gelesen wird. Ohne dieses B zu verändern geht M einen Schritt nach links und geht in den Zustand q_1 über. Die Bandzelle über der der Lese/Schreibkopf nun positioniert ist, enthält die am weitesten rechts liegende Binärziffer der Eingabe. (Bisher wurde am Eingabestring nichts verändert.) Im Zustand q_1 wird diese Binärziffer nun dualisiert – also 0 mit 1, bzw. 1 mit 0 überschrieben – und mit einem Schritt nach links in den Zustand q_2 übergegangen. Im Zustand q_2 geht M nun soweit nach links bis ein B gelesen wird. Ohne dieses B zu ändern geht M nun einen Schritt nach rechts und in den Endzustand q_3 . Der Lese/Schreib-Kopf zeigt auf das erste Symbol des Binärstrings, der somit nun der Ausgabestring ist.
-

$$f_M(n) = \begin{cases} n + 1 & \text{wenn } n \text{ gerade ist,} \\ n - 1 & \text{wenn } n \text{ ungerade ist.} \end{cases}$$

Aufgabe 2.2

- Geben Sie ein 2-RM Programm P an, das die Funktion $f_P(n) = 3n + 1$ berechnet.
- Geben Sie eine vollständige Rechnung (Konfigurationsfolge) für die Eingabe 1 an.
- Beschreiben Sie die Arbeitsweise von P für eine beliebige Eingabe $n \in \mathbb{N}$ in Worten.

Lösung 2.2

- $P = (0, \{(0, t_1, 5, 1), (1, S_1, 2), (2, A_2, 3), (3, A_2, 4), (4, A_2, 0), (5, t_2, 8, 6), (6, S_2, 7), (7, A_1, 5), (8, A_1, 9)\})$
- $0:(1, 0) \Rightarrow 1:(1, 0) \Rightarrow 2:(0, 0) \Rightarrow 3:(0, 1) \Rightarrow 4:(0, 2) \Rightarrow 0:(0, 3) \Rightarrow 5:(0, 3) \Rightarrow 6:(0, 3)$
 $\Rightarrow 7:(0, 2) \Rightarrow 5:(1, 2) \Rightarrow 6:(1, 2) \Rightarrow 7:(1, 1) \Rightarrow 5:(2, 1) \Rightarrow 6:(2, 1) \Rightarrow 7:(2, 0)$
 $\Rightarrow 5:(3, 0) \Rightarrow 8:(3, 0) \Rightarrow 9:(4, 0)$

Nach Ausführung des Befehls mit der Kennmarke 8 hält P , da 9 keine Kennmarke und somit eine Endmarke ist. Das Programm hält mit der Ausgabe 4. (Beachten Sie, dass das Ergebnis immer in $R(1)$ stehen muss. Der Inhalt anderer Register ist für das Ergebnis nicht relevant.)

- Laut Definition gilt anfangs in $R(1) = n$ und $R(2) = 0$. Solange $R(1)$ (noch) nicht leer ist, wird $R(1)$ jeweils um 1 vermindert und $R(2)$ um 3 erhöht (= Schleife 0–4). Daher gilt $R(1) = 0$ und $R(2) = 3n$ sobald die Rechnung an der Kennmarke 5 angelangt ist. Danach wird solange $R(2)$ um 1 vermindert und $R(1)$ um 1 erhöht, bis $R(2)$ leer ist und somit nun $R(1) = 3n$ gilt (= Schleife 5–7). Danach wird $R(1)$ an der Kennmarke 8 um 1 erhöht, sodass nun $R(1) = 3n + 1$ gilt.

Aufgabe 2.3

- Geben Sie einen λ -Term **imp** an, der die Wahrheitsfunktion der Implikation darstellt. Gehen Sie dabei wie folgt vor: Instanzieren Sie den Term für **if_then_else** auf Folie 36 (Foliensatz 2) auf geeignete Weise. Durch anschließende β - und α -Reduktion ergibt sich der gesuchte Kombinator **imp**. (Geben Sie alle Schritte der Konstruktion explizit an.) Geben Sie **imp** schließlich auch mit voller Klammerung (also ohne Schreibkonventionen) an.
- Werten Sie **imp** für das Argumentpaar **T F** schrittweise aus.
- Werten Sie **imp** für das Argumentpaar **F F** schrittweise aus.
- Werten Sie **imp** für das Argumentpaar **T T** schrittweise aus.

Unterstreichen Sie den jeweils verwendeten Redex in allen β -Reduktionen.

Lösung 2.3

- Zur Erinnerung: **T** = $\lambda xy.x$, **F** = $\lambda xy.y$, **if_then_else** = $\lambda pxy.pxy$.

Informell, in gewohnter Infix-Schreibweise, gilt $X \text{ imp } Y = \text{if } X \text{ then } Y \text{ else } \mathbf{T}$. Daher

$$\begin{aligned}
 \mathbf{imp} &= \lambda xy.(\mathbf{if_then_else})xy\mathbf{T} \\
 &\rightarrow_{\alpha} \lambda xy.(\lambda px'y'.px'y')xy\mathbf{T} \\
 &\rightarrow_{\beta} \lambda xy.(\lambda x'y'.xx'y')y\mathbf{T} \\
 &\rightarrow_{\beta} \lambda xy.(\lambda y'.xyy')\mathbf{T} \\
 &\rightarrow_{\beta} \lambda xy.xy\mathbf{T} \\
 &\rightarrow_{\alpha} \lambda xy.xy\lambda x'y'.x'
 \end{aligned}$$

Der λ -Term in der letzten Zeile ist in Normalform und ein Kombinator.

Wir erhalten also **imp** = $\lambda xy.xy\mathbf{T}$. (Es ist geschickt, **T** erst in entsprechenden Berechnungen zu expandieren.)

In voller Klammerung (und expandiert) gilt **imp** = $(\lambda x.(\lambda y.((xy)(\lambda x'.(\lambda y'.x')))))$.

b)

$$\begin{aligned}
 \text{impTF} &= \frac{(\lambda xy. xy \mathbf{T}) \mathbf{TF}}{(\lambda y. \mathbf{T} y \mathbf{T}) \mathbf{F}} \\
 &\rightarrow_{\beta} \frac{\mathbf{TFT}}{\mathbf{TFT}} \\
 &= \frac{(\lambda xy. x) \mathbf{FT}}{(\lambda y. \mathbf{F}) \mathbf{T}} \\
 &\rightarrow_{\beta} \mathbf{F}
 \end{aligned}$$

c)

$$\begin{aligned}
 \text{impFF} &= \frac{(\lambda xy. xy \mathbf{T}) \mathbf{FF}}{(\lambda y. \mathbf{F} y \mathbf{T}) \mathbf{F}} \\
 &\rightarrow_{\beta} \frac{\mathbf{FFT}}{\mathbf{FFT}} \\
 &= \frac{(\lambda xy. y) \mathbf{FT}}{(\lambda y. y) \mathbf{T}} \\
 &\rightarrow_{\beta} \mathbf{T}
 \end{aligned}$$

d)

$$\begin{aligned}
 \text{impTT} &= \frac{(\lambda xy. xy \mathbf{T}) \mathbf{TT}}{(\lambda y. \mathbf{T} y \mathbf{T}) \mathbf{T}} \\
 &\rightarrow_{\beta} \frac{\mathbf{TTT}}{\mathbf{TTT}} \\
 &= \frac{(\lambda xy. x) \mathbf{TT}}{(\lambda y. \mathbf{T}) \mathbf{T}} \\
 &\rightarrow_{\beta} \mathbf{T}
 \end{aligned}$$

Aufgabe 2.4

- Präsentieren Sie einen **Pr**-Funktionsausdruck E für die Funktion $E(m, n) = (n + 6)^{2m+1}$. Verwenden Sie dabei die bereits in der Vorlesung definierten Ausdrücke Add und $Mult$.
- Werten Sie Ihren Ausdruck E für die Eingabe $(3, 4)$ schrittweise aus. Behandeln Sie dabei Add und $Mult$ wie Grundfunktionen. (Übersetzen Sie also $Add(\cdot, \cdot)$ und $Mult(\cdot, \cdot)$ direkt in entsprechende Summen bzw. Produkte.)
- Analysieren Sie die Funktion $\bar{\mu}E$. Geben Sie zwei verschiedene Zahlenpaare (m, n) an, für die $\bar{\mu}E(m, n)$ definiert ist, sowie zwei weitere Zahlenpaare, für die $\bar{\mu}E(m, n)$ undefiniert ist.
- Welche Funktion wird durch $\mu_0 E$ dargestellt?

Lösung 2.4

- $E(m, n) = (n + 6)^{2m+1}$ lässt sich durch die folgenden Rekursionsgleichungen definieren:

Anfang: $E(0, n) = n + 6$

Übergang: $E(m + 1, n) = E(m, n) \cdot (n + 6)^2$.

Daher lässt sich E durch den Ausdruck $\mathbf{Pr}(Add \circ (P_1^1, C_6^1), Mult \circ (Qu \circ (Add \circ (P_3^3, C_6^3)), P_2^3))$ darstellen, wobei Qu eine Abkürzung für $(Mult \circ (P_1^1, P_1^1))$ ist.

- Wir werten jeweils $E(m, 4)$ für $m = 0, \dots, 3$ aus:

$$E(0, 4) = Add \circ (P_1^1, C_6^1)(4) = Add(4, 6) = 4 + 6 = 10.$$

$$\begin{aligned}
 E(1, 4) &= Mult \circ (Qu \circ (Add \circ (P_3^3, C_6^3)), P_2^3)(0, E(0, 4), 4) = \\
 &Mult \circ (Qu \circ Add(4, 6), E(0, 4)) = (4 + 6)^2 \cdot 10 = 10^2 \cdot 10 = 1000.
 \end{aligned}$$

$$\begin{aligned}
 E(2, 4) &= Mult \circ (Qu \circ (Add \circ (P_3^3, C_6^3)), P_2^3)(1, E(1, 4), 4) = \\
 &Mult \circ (Qu \circ Add(4, 6), E(1, 4)) = (4 + 6)^2 \cdot 1000 = 10^2 \cdot 1000 = 100000.
 \end{aligned}$$

$$E(3, 4) = Mult \circ (Qu \circ (Add \circ (P_3^3, C_6^3)), P_2^3)(2, E(2, 4), 4) = \\ Mult \circ (Qu \circ Add(4, 6), E(2, 4)) = (4 + 6)^2 \cdot 100000 = 10^2 \cdot 100000 = 10000000.$$

- c) Es gilt $\bar{\mu}E(m, n) = \min_y [E(y, n) = m]$. Wir suchen also kleinste (natürlich-zahlige) Lösungen für y von $(n + 6)^{2y+1} = m$ bzw. Paare (m, n) für die keine solche Lösung existiert.

Für $m = 6$ und $n = 0$ ist $y = 0$ eine Lösung, da $(0 + 6)^{2 \cdot 0 + 1} = 6$ gilt. Somit erhalten wir $\bar{\mu}E(6, 0) = 0$.

Für $m = 1000$ und $n = 4$ ist $y = 1$ eine Lösung, da $(4 + 6)^{2 \cdot 1 + 1} = 10^3 = 1000$ gilt. Somit erhalten wir $\bar{\mu}E(1000, 4) = 1$.

Hingegen sind, z.B., $E(0, 0)$ und $E(0, 1)$ undefiniert.

- d) Es gilt $\mu_0 E = \min_y [E(y, n) = 0]$. Für keine $n, y \in \mathbb{N}$ gilt $(n + 6)^{2y+1} = 0$. Daher stellt $\mu_0 E$ eine nirgends definierte 1-stellige Funktion dar.

Aufgabe 2.5

Erklären Sie für jedes der folgenden Entscheidungsprobleme, warum der Satz von Rice (in jeweils einer der beiden Varianten) anwendbar bzw. nicht anwendbar ist um zu folgern, dass das Problem unentscheidbar ist. Falls der Satz von Rice anwendbar ist, geben Sie die entsprechende Sprach- bzw. Funktions-Eigenschaft an.

- Berechnet eine vorgelegte TM die selbe Funktion wie der Funktionsausdruck $\mathbf{Pr}(P_1^1, S \circ P_2^3)$?
- Berechnet eine vorgelegte TM eine durch einen $\mathbf{Pr}$ -Funktionsausdruck darstellbare Funktion?
- Berechnet eine vorgelegte TM eine Funktion, die man mit μ_0 , $\mathbf{Pr}$ und $\circ$, angewendet auf Grundfunktionen, darstellen kann?
- Verändert eine vorgelegte TM bei leerem Eingabestring das Band während der Berechnung an mehr als 100 Stellen?
- Besteht eine (als entsprechende TM) vorgelegte Sprache aus mehr als 100 Wörtern?
- Ist eine (als entsprechende TM) vorgelegte Sprache rekursiv?

Lösung 2.5

- Der Funktionsausdruck stellt eine bestimmte Funktion f dar. (In diesem Fall die Identitätsfunktion $f(n) = n$, was aber für das Argument keine Rolle spielt.) Der Satz von Rice für Funktionen ist also — mit Verweis auf die Funktions-Eigenschaft $\{f\}$ — anwendbar. Entsprechend ist das Problem unentscheidbar.
- Dieses Problem ist ebenfalls nach dem Satz von Rice für Funktionen unentscheidbar. Die entsprechende Funktions-Eigenschaft ist die Menge der primitiv-rekursiven Funktionen.
- Die Menge der Funktionen, die man mit μ_0 , $\mathbf{Pr}$ und $\circ$ angewendet auf Grundfunktionen darstellen kann, bezeichnet die partiell rekursiven Funktionen und somit eine triviale Eigenschaft berechenbarer Funktionen. (Jede berechenbare Funktion ist partiell rekursiv). Der Satz von Rice ist also nicht anwendbar; das Problem ist (trivial) entscheidbar.
- Die Menge der von einer TM mit dieser Eigenschaft akzeptierten Sprachen bzw. der berechneten Funktionen wird hier nicht extensional, sondern intensional beschrieben. Der Satz von Rice ist also nicht anwendbar.
- Es geht um die (extensionale) Sprach-Eigenschaft $\{\mathcal{L} \mid |\mathcal{L}| > 100\}$. Der Satz von Rice (für Sprachen) ist also anwendbar um zu folgern, dass das Problem unentscheidbar ist.

- f) Die Menge der rekursiven (= entscheidbaren) Sprachen ist eine echte, nicht-leere Teilmenge der rekursiv aufzählbaren (= semi-entscheidbaren) Sprachen. Es handelt sich also um eine nicht-triviale Sprach-Eigenschaft. Somit ist der Satz von Rice anwendbar um zu folgern, dass das Problem unentscheidbar ist.

Aufgabe 2.6

Beurteilen Sie die Korrektheit jeder der folgenden Aussagen. Begründen Sie jeweils Ihre Antwort. Beachten Sie, dass Begründungen in manchen Fällen konkrete Berechnungen verlangen.

- Zu jeder totalen Funktion h vom Typ $\mathbb{N}^2 \rightarrow \{0, 1\}$ gibt es eine Turingmaschine, die die Funktion h berechnet.
- Für die vom 3-RM-Programm $(0, \{(0, A_2, 1), (1, S_1, 2), (2, t_1, 3, 0)\})$ berechnete Funktion g vom Typ $\mathbb{N} \rightarrow \mathbb{N}$ gilt $g(2) = 2$.
- Für $f(x, y) = x^2 + y$ vom Typ $\mathbb{N}^2 \rightarrow \mathbb{N}$ gilt: $(\mu_0 f)(1)$ ist undefiniert; $(\bar{\mu} f)(2, 1) = 1$; $(\bar{\mu} f)(3, 1)$ ist undefiniert.
- Aus dem Satz von Rice folgt, dass man nicht entscheiden kann, ob eine vorgelegte Turingmaschine entweder die Funktion g aus b) oder die Funktion f aus c) berechnet.

Lösung 2.6

- a) Diese Aussage ist falsch, da es überabzählbar viele totale Funktionen dieses Typs gibt, aber nur abzählbar viele Turingmaschinen.

Ein alternative Begründung lautet:

Jede Menge A von Paaren natürlicher Zahlen ist als charakteristische und somit totale Funktion f_A vom Typ $\mathbb{N}^2 \rightarrow \{0, 1\}$ wie folgt repräsentierbar: $f_A(m, n) = 1 \Leftrightarrow (m, n) \in A$ und $f_A(m, n) = 0 \Leftrightarrow (m, n) \notin A$. Die Aussage ist daher äquivalent zur offensichtlich falschen Behauptung, dass jede Menge von paaren natürlicher Zahlen entscheidbar (= rekursiv) ist.

- b) Da eine Funktion vom Typ $\mathbb{N} \rightarrow \mathbb{N}$ berechnet wird, gilt anfangs $R(1) = 2$, $R(2) = 0$ und $R(3) = 0$. Die entsprechende Berechnung lautet

$$0:(2, 0, 0) \Rightarrow 1:(2, 1, 0) \Rightarrow 2:(1, 1, 0) \Rightarrow 0:(1, 1, 0) \Rightarrow 1:(1, 2, 0) \Rightarrow 2:(0, 2, 0) \Rightarrow 3:(0, 2, 0).$$

An der Endmarke 3 gilt $R(1) = 0$ und daher $g(2) = 0$. ($R(2)$ und $R(3)$ sind irrelevant, da g vom Typ $\mathbb{N} \rightarrow \mathbb{N}$ ist; siehe Foliensatz 2, Folie 23) Die Aussage ist also falsch.

- c) Es gilt $(\mu_0 f)(1) = \min_{k \geq 0} [k^2 + 1 = 0]$. $(\mu_0 f)(1)$ ist daher undefiniert.

Es gilt $(\bar{\mu} f)(x, y) = \min_{k \geq 0} [k^2 + y = x]$ und daher

$$\bar{\mu} f(x, y) = \begin{cases} \sqrt{x - y} & \text{falls } x - y \text{ eine Quadratzahl (in } \mathbb{N}) \text{ ist,} \\ \text{undefiniert} & \text{sonst} \end{cases}$$

Es gilt daher $(\bar{\mu} f)(2, 1) = \sqrt{2 - 1} = 1$. Andererseits ist $(\bar{\mu} f)(3, 1)$ undefiniert, da $3 - 1 = 2$ keine Quadratzahl ist. Die gesamte Aussage ist also korrekt.

- d) Es geht um die (extensional definierte) nicht-triviale Funktions-Eigenschaft $\{f, g\}$. Der Satz von Rice für Funktionen ist daher anwendbar um zu folgern, dass das Problem tatsächlich unentscheidbar ist.

Aufgabe 2.7

Gegeben sei folgende (deterministische) Turingmaschine M (Variante aus Foliensatz 3, Folien 11 ff):

$$M = (\{q_0, q_1, q_2\}, \{a, b, c\}, \{A, Z_0, B\}, \delta, q_0, \{Z_0 Z_1, Z_2\}, B, \{q_2\})$$

wobei die Übergangsfunktion δ folgendermaßen definiert ist:

- 1 : $\delta(q_0, a, B) = (q_0, A, R, R)$
- 2 : $\delta(q_0, b, B) = (q_0, A, R, R)$
- 3 : $\delta(q_0, c, B) = (q_1, B, R, L)$
- 4 : $\delta(q_1, a, A) = (q_1, B, R, L)$
- 5 : $\delta(q_1, b, A) = (q_1, B, R, L)$
- 6 : $\delta(q_1, Z_2, Z_0) = (q_2, Z_0, S, R)$

- a) Beschreiben Sie kurz die Arbeitsweise dieser Maschine.
- b) Geben Sie $L = L(M)$ (also die Sprache, die von M akzeptiert wird) an.
- c) Handelt es sich bei M um einen LBA, Kellerautomaten und/oder EA? Begründen Sie Ihre Antwort.
- d) Wo in der Chomsky-Hierarchie ordnen Sie die Sprache L ein? Begründen Sie Ihre Antwort.

Lösung 2.7

- a) 1, 2, 3 : Für jedes eingelesene Symbol a oder b wird ein Symbol A auf das Arbeitsband geschrieben. Dies geschieht solange, bis ein Symbol c im Eingabewort erreicht wird (Übergang 3).
- 4, 5 : nun wird für jedes weitere eingelesene Symbol a oder b ein Symbol A dem Arbeitsband gelöscht.
- 6 : wird gleichzeitig mit dem Ende der Eingabe (dem Erreichen von Z_2 auf dem Eingabeband) das linke Begrenzungssymbol Z_0 auf dem Arbeitsband erreicht, geht M in den (einzigen) Endzustand q_2 über und akzeptiert somit die Eingabe.
- b) $L = \{wcv \mid w, v \in \{a, b\}^*, |w| = |v|\}$
(Wurde Transition 5 wie ursprünglich angegeben als $\delta(q_1, b, B) = (q_1, B, R, L)$ angenommen, so wäre entsprechend $L = \{wca^{|w|} \mid w \in \{a, b\}^*\}$.)
- c) M ist ein LBA, da auf dem Arbeitsband während der Analyse eines Wortes w weniger als $|w|$ Zellen verwendet werden. M erfüllt darüberhinaus die Kellerautomatenbedingung, da zu jedem Zeitpunkt immer nur das "oberste" Symbol auf dem Arbeitsband gelesen wird.
- d) Die Sprache L ist jedenfalls kontextfrei, da sie von einem Kellerautomaten akzeptiert wird. Damit ist L entsprechend der Chomsky Hierarchie natürlich auch kontextsensitiv und rekursiv aufzählbar. L ist aber sicher nicht regulär, was z.B. mit dem Pumping Lemma für reguläre Sprachen gezeigt werden kann, ähnlich der Folie 51 mit z.B. dem Wort $w = a^m c b^m$.

Aufgabe 2.8

Sind folgende Aussagen korrekt? Begründen Sie jeweils Ihre Antwort.

- a) Sei $A = \{a^n b^n \mid n \geq 0\}$ und $B = \{c^n d^n \mid n \geq 0\}$. Dann gilt: $A \cdot B = \{a^n b^n c^n d^n \mid n \geq 0\}$.
- b) Das Komplement kontextsensitiver Sprachen ist immer unendlich.
- c) Jede formale Sprache, deren Komplement endlich ist, ist kontextfrei.
- d) Für jede beliebige formale Sprache L gilt: L^+ enthält nicht das Leerwort.

- e) Jede Teilmenge einer regulären Sprache ist selbst auch regulär.

Lösung 2.8

- a) **Falsch.** Das ist sicher nicht korrekt, da z.B. $a^2b^2 \in A$ und $c^3d^3 \in B$, aber $a^2b^2c^3d^3 \notin \{a^n b^n c^n d^n \mid n \geq 0\}$.
- (Richtig wäre in diesem Fall: $\{a^n b^n \mid n \geq 0\} \cdot \{c^n d^n \mid n \geq 0\} = \{a^n b^n c^k d^k \mid n, k \geq 0\}$. Beachten Sie auch, dass A und B kontextfrei sind, und kontextfreie Sprachen unter Konkatenation abgeschlossen sind. Die Sprache $\{a^n b^n c^n d^n \mid n \geq 0\}$ ist hingegen kontextsensitiv, aber sicher nicht kontextfrei.)
- b) **Falsch.** Hat eine Sprache L ein endliches Komplement, so ist L regulär. Reguläre Sprachen sind aber eine echte Teilmenge der kontextsensitiven Sprachen (Chomsky Hierarchie). Es gibt also durchaus kontextsensitive Sprachen, die ein endliches Komplement haben.
- c) **Richtig.** Hat eine Sprache L ein endliches Komplement, so ist L regulär und damit auch kontextfrei, da reguläre Sprachen eine echte Teilmenge der kontextfreien Sprachen sind (Chomsky Hierarchie).
- d) **Falsch.** Gilt $\varepsilon \in L$, so auch $\varepsilon \in L^+$.
- e) **Falsch.** Gegenbeispiel: $L_1 = \{a^n b^n \mid n \geq 0\}$ und $L_2 = \{a, b\}^*$. Dann gilt: $L_1 \subseteq L_2$, L_1 ist aber nicht regulär.

Aufgabe 2.9

Die unten gegebenen Sprachen A und B sind nicht regulär, was mit dem Pumping Lemma für reguläre Sprachen gezeigt werden soll. Dabei haben sich jedoch Fehler in der Argumentation eingeschlichen. Beschreiben Sie diese Fehler, und verbessern Sie die Beweise entsprechend.

- a) Sei $A = \{(a^n b^n)^3 \mid n \geq 0\}$.

Wir beweisen mit Hilfe des Pumping Lemmas für reguläre Sprachen, dass L nicht regulär ist:

Beweis indirekt. Angenommen, A ist regulär. Sei dann m die Konstante aus dem Pumping Lemma und $w = (a^m b^m)^3$. Dann gilt $w \in L$ und $|w| = 6m$, also $|w| > m$.

Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Nachdem $|xy| \leq m$ und $w = (a^m b^m)^3$, kann xy nur aus (mindestens 1 bis höchstens m) Symbolen a am Wortanfang bestehen.

Nach dem Pumping Lemma muss nun $xy^i z \in L$ für alle $i \geq 0$ gelten.

Bei $i = 2$ müsste (wenn L regulär wäre) auch $xy^2 z = (a^m b^m)^{3-|y|}$ aus L sein. Dies ist hier aber nicht der Fall, da nun mindestens ein Symbol a am Wortanfang fehlt, und damit die Anzahl der Symbole a in w nicht mehr gleich gross ist wie die Anzahl der Symbole b . Dementsprechend ist das Wort $xy^2 z$ nicht in der Sprache A ! Wir haben also einen Widerspruch gefunden, A kann somit keine reguläre Sprache sein.

- b) Sei $B = \{a^n b^m \mid n \geq m\}$.

Wir beweisen mit Hilfe des Pumping Lemmas für reguläre Sprachen, dass L nicht regulär ist:

Beweis indirekt. Angenommen, L ist regulär. Sei dann m die Konstante aus dem Pumping Lemma und $w = a^m b^m$. Dann gilt $w \in L$ und $|w| = 2m \geq m$.

w kann nun in xyz so aufgeteilt werden, dass $|xy| \leq m$ und $|y| > 0$. Nachdem $|xy| \leq m$ und $w = a^m b^m$, kann xy nur aus Symbolen a bestehen. Nach dem Pumping Lemma muss aber $xy^i z \in L$ für alle $i \geq 0$ gelten.

Für $i = 2$ müsste also auch $xy^2z = a^{m+|y|}b^m$ aus L sein, was aber nicht der Fall ist. Wir haben einen Widerspruch gefunden, B kann somit keine reguläre Sprache sein.

Lösung 2.9

- a) Der Beweis ist weitestgehend korrekt, allerdings gilt nicht $xy^2z = (a^mb^m)^{3-|y|}$ und es fehlte(n) auch kein(e) Symbol(e) a . Richtigerweise müsste es lauten: $xy^2z = a^{m+|y|}b^m(a^mb^m)^2$. Damit kommt nun mindestens ein Symbol a am Wortanfang hinzu.

Darüberhinaus wird die Sprache A mitunter auch als L bezeichnet. Konsistenterweise sollte also L durch A ersetzt werden.

- b) Hier ist die Wahl von i problematisch. Denn mit $i = 2$ wird die Anzahl der Symbole a in w_i erhöht. Damit liegt nun allerdings die aufgepumpte Variante von w wiederum in B . Es wird also kein Widerspruch herbeigeführt.

Zielführend wäre hier $i = 0$ zu wählen. Denn nur dann fällt mindestens ein Symbol a aus w weg, womit nun aber definitiv weniger Symbole a als b im Wort vorkommen. Erst damit ergibt sich der gewünschte Widerspruch.

Darüberhinaus wird die Sprache B mitunter auch als L bezeichnet. Aus Gründen der Konsistenz sollte hier also L durch B ersetzt werden.

Aufgabe 2.10

Beweisen Sie mit Hilfe des Pumping Lemmas für reguläre Sprachen, dass die folgenden Sprachen L nicht regulär sind.

Geben Sie darüberhinaus auch an, ob es sich bei L jeweils um eine kontextfreie Sprache handelt, und begründen Sie Ihre Antwort.

- a) $L = \{xyy^rx^r \mid x, y \in \{a, b, c\}^*\}$
 b) $L = \{xyx^ry^r \mid x, y \in \{a, b, c\}^*\}$

Lösung 2.10

- a) Beweis indirekt. Angenommen, L ist regulär. Sei dann m die Konstante aus dem Pumping Lemma und das gewählte Wort z.B.

$$w = a^mcca^m.$$

Dann gilt $w \in L$ und $|w| = 2m + 2 > m$.

Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Nachdem $|xy| \leq m$ und $w = a^mcca^m$, kann xy nur aus (mindestens einem bis höchstens m) Symbolen a der ersten Worthälfte bestehen.

Nach dem Pumping Lemma muss aber $xy^iz \in L$ für alle $i \geq 0$ gelten.

Wählen wir nun $i = 0$, so müsste auch $xy^0z = a^{m-|y|}cca^m$ aus L sein, was aber nicht der Fall ist! Da y aus mindestens einem Symbol besteht, fällt mit $i = 0$ auch mindestens ein Symbol a am Wortanfang weg. Damit ist im resultierenden Wort aber die zweite Worthälfte nicht mehr das Spiegelbild der ersten. Widerspruch! L kann somit keine reguläre Sprache sein.

L entspricht der Sprache $\{ww^r \mid w \in \{a, b, c\}^*\}$, also der Menge der Palindrome über dem Alphabet $\{a, b, c\}$ mit gerader Wortlänge.

L ist **kontextfrei**, und wird z.B. von folgender kontextfreien Grammatik erzeugt:

$$G = (\{S\}, \{a, b, c\}, \{S \rightarrow xSx \mid x \in \{a, b, c\}\} \cup \{S \rightarrow \varepsilon\}, S)$$

- b) Beweis indirekt. Angenommen, L ist regulär. Sei dann m die Konstante aus dem Pumping Lemma und das gewählte Wort z.B.

$$w = a^m c a^m c.$$

Dann gilt $w \in L$ und $|w| = 2m + 2$, also jedenfalls $|w| \geq m$.

Nun kann w in xyz so aufgeteilt werden, dass $|xy| \leq m$ und $|y| > 0$. Nachdem $|xy| \leq m$ und $w = a^m c a^m c$, kann xy nur aus (mindestens einem bis maximal m) Symbolen a am Wortanfang bestehen.

Nach dem Pumping Lemma muss dann $xy^i z \in L$ für alle $i \geq 0$ gelten.

Für z.B. $i = 2$ müsste nun aber auch $xy^2 z = a^{m+|y|} c a^m c$ ein Wort in L sein. Dies ist aber nicht der Fall, da nun vor dem ersten Symbol c mindestens ein Symbol a mehr vorkommt, als im nachfolgenden a -Block, und das resultierende Wort damit nicht in L sein kann.

Widerspruch! L kann keine reguläre Sprache sein.

L ist **nicht kontextfrei**, was man beispielsweise mit dem Pumping Lemma für kontextfreie Sprachen zeigen könnte (ähnlich dem Beweis auf Folie(n) 94 ff mit z.B. dem Wort $w = a^m b^m a^m b^m$)

L ist aber kontextsensitiv und könnte von einem linear beschränkten Automaten akzeptiert werden, bzw. von einer kontextsensitiven oder monotonen Grammatik erzeugt werden.

Aufgabe 2.11

Zeigen Sie die Kontextfreiheit der folgenden Sprachen L über $\Sigma = \{a, b, c\}$ mit Hilfe des Satzes von Chomsky-Schützenberger (indem Sie jeweils entsprechende Sprachen D_n und R sowie einen entsprechenden Homomorphismus h so angeben, dass gilt: $L = h(D_n \cap R)$).

Geben Sie darüberhinaus jeweils eine kontextfreie Grammatik an, welche die angegebene Sprache erzeugt.

- a) $\{xx^r yy^r \mid x, y \in \{a, b\}^+\}$
- b) $\{wc^n \mid w \in \{a, b\}^*, |w| \geq n\}$

Lösung 2.11

- a) L ist kontextfrei, da $L = h(D_4 \cap R)$, wobei

$$R = \{(1, [1]^+ \{ \}_1,]_1\}^+ \{(2, [2]^+ \{ \}_2,]_2\}^+$$

und

$$h : \{(1,)_1, [1,]_1, (2,)_2, [2,]_2\}^* \longrightarrow \{a, b\}^* \quad \text{mit}$$

$$h((1) = h(())_1 = h((2) = h(())_2 = a, \quad h([1) = h(]_1) = h([2) = h(]_2) = b$$

L wird z.B. von der folgenden kontextfreien Grammatik erzeugt:

$$G = (\{S, A\}, \{a, b\}, P, S) \text{ mit } P = \{S \rightarrow AA, \quad A \rightarrow aAa \mid bAb \mid aa \mid bb\}$$

- b) L ist kontextfrei, da $L = h(D_4 \cap R)$, wobei

$$R = \{(1, (2, (3, (4)^* \{ \}_1,)_2,)_3,)_4\}^*$$

und

$$h : \{(1,)_1, (2,)_2, (3,)_3, (4,)_4\}^* \longrightarrow \{a, b, c\}^* \quad \text{mit}$$

$$h((1) = h((3) = a, \quad h((2) = h((4) = b, \quad h(())_1 = h(())_2 = c, \quad h(())_3 = h(())_4 = \varepsilon$$

L wird z.B. von der folgenden kontextfreien Grammatik erzeugt:

$$G = (\{S\}, \{a, b, c\}, P, S) \text{ mit } P = \{S \rightarrow aSc \mid bSc \mid aS \mid bS \mid \varepsilon\}$$

Aufgabe 2.12

Geben Sie für jede der folgenden Grammatiken an, ob diese regulär, kontextfrei, monoton, und/oder kontextsensitiv ist. Geben Sie weiters an, welche Sprache von der jeweils angegebenen Grammatik erzeugt wird, und ob diese regulär, kontextfrei, kontextsensitiv und/oder rekursiv aufzählbar ist. Begründen Sie Ihre Antwort.

Beispiel:

$$\text{Sei } G_{BSP} = (\{S\}, \{a, b\}, \{S \rightarrow Sxx \mid x \in \{a, b\}\} \cup \{S \rightarrow \varepsilon\}, S).$$

G_{BSP} ist kontextfrei. Wegen z.B. der Produktion $S \rightarrow Saa$ ist G_{BSP} nicht regulär, und wegen $S \rightarrow \varepsilon$ nicht monoton und auch nicht kontextsensitiv, nachdem S auch auf der rechten Seite von Produktionen vorkommt.

$L(G_{BSP}) = \{aa, bb\}^*$ ist aber regulär, und damit auch kontextfrei, kontextsensitiv und rekursiv aufzählbar.

- a) $G = (\{S\}, \{a, b, c\}, \{S \rightarrow xSx \mid x \in \{a, b, c\}\} \cup \{S \rightarrow \varepsilon\}, S)$
- b) $G = (\{S, A, B\}, \{a, b, c\}, \{S \rightarrow A \mid B, \quad A \rightarrow aA \mid a, \quad B \rightarrow bBc \mid bB \mid b\}, S)$
- c) $G = (\{S, B\}, \{a, b\}, \{S \rightarrow aba \mid aSBa, \quad aB \rightarrow Ba, \quad bB \rightarrow bb\}, S)$

Lösung 2.12

- a) G ist kontextfrei, da auf der linken Seite der Produktionen jeweils immer nur ein Nonterminalsymbol vorkommt. Wegen z.B. $S \rightarrow aSa$ ist G nicht regulär, wegen $S \rightarrow \varepsilon$ ist G weder monoton noch kontextsensitiv, da S auch auf der rechten Seite einer Produktion (z.B. $S \rightarrow cSc$) vorkommt.

$L(G) = \{ww^r \mid w \in \{a, b, c\}^*\}$ ist eine kontextfreie Sprache, und somit auch kontextsensitiv und rekursiv aufzählbar (Chomsky Hierarchie).

- b) G ist wegen z.B. $B \rightarrow bBc$ nicht regulär. G ist kontextfrei, da auf der linken Seite jeder Produktion jeweils genau ein Nonterminal steht. G ist darüberhinaus kontextsensitiv und auch monoton, da für alle Produktionen $\alpha \rightarrow \beta$ in G gilt, dass $|\alpha| \leq |\beta|$.

$L(G) = \{a^n \mid n \geq 1\} \cup \{b^n c^k \mid n > k \geq 0\}$. Diese Sprache ist nicht regulär (siehe z.B. Aufgabe 9). Sie ist aber kontextfrei und damit auch, aufgrund der Chomsky Hierarchie, kontextsensitiv und rekursiv aufzählbar.

- c) G ist monoton und unbeschränkt, aber wegen z.B. $aB \rightarrow Ba$ weder regulär, noch kontextfrei, noch kontextsensitiv.

$L(G) = \{a^n b^n a^n \mid n \geq 1\}$ ist kontextsensitiv und daher auch rekursiv aufzählbar. (aber weder regulär noch kontextfrei, siehe z.B. Folie 94 ff.).