Institut für Theoretische Informatik
Lehrstuhl Prof. Dr. D. Wagner

Übungsblatt 7

Vorlesung Theoretische Grundlagen der Informatik im WS 17/18

Ausgabe 24. Januar 2018
Abgabe 6. Februar 2018, 11:00 Uhr (im Kasten im UG von Gebäude 50.34)

Aufgabe 1

(1 + 2 = 3 Punkte)

Sei $G = (V, \Sigma, S, R)$ mit $V = \{A, B, C, D, E\}$ und $\Sigma = \{a, b\}$ die durch folgende Regelmenge gegebene Grammatik:

$$\begin{aligned} S &\to AB \mid AC \mid DD \\ A &\to a \\ B &\to b \\ C &\to SB \mid SC \\ D &\to AE \mid EB \mid E \\ E &\to DA \mid DB \end{aligned}$$

(a) Enthält die Grammatik nutzlose Symbole? Wenn ja, welche?

(b) Ist die von der Grammatik erzeugte Sprache endlich? Begründen Sie Ihre Antwort.
Hinweis: Die Grammatik muss dazu nicht notwendigerweise in Chomsky-Normalform gebracht werden.

Lösung:

(a) Berechne zunächst erzeugende Symbole: A, B, S, C. Berechne anschließend nützliche Symbole ausgehend von den erzeugenden Symbolen: Alle erzeugenden Symbole sind auch nützlich. Nutzlose Symbole sind also D, E. (Nützliche Symbole können z.B. durch Tiefensuche mit Wurzel S aus dem durch die erzeugenden Symbole und erzeugenden Regeln induzierten Abhängigkeitsgraphen ermittelt werden.)

(b) Betrachte Abhängigkeitsgraphen der durch die nützlichen Symbole und Regeln induziert wird. Dieser enthält zwei Zyklen $S \to C \to S$ und $C \to C$. Da die Zyklen nicht ausschließlich durch Kettenregeln induziert werden, ist die von der Grammatik erzeugte Sprache unendlich.

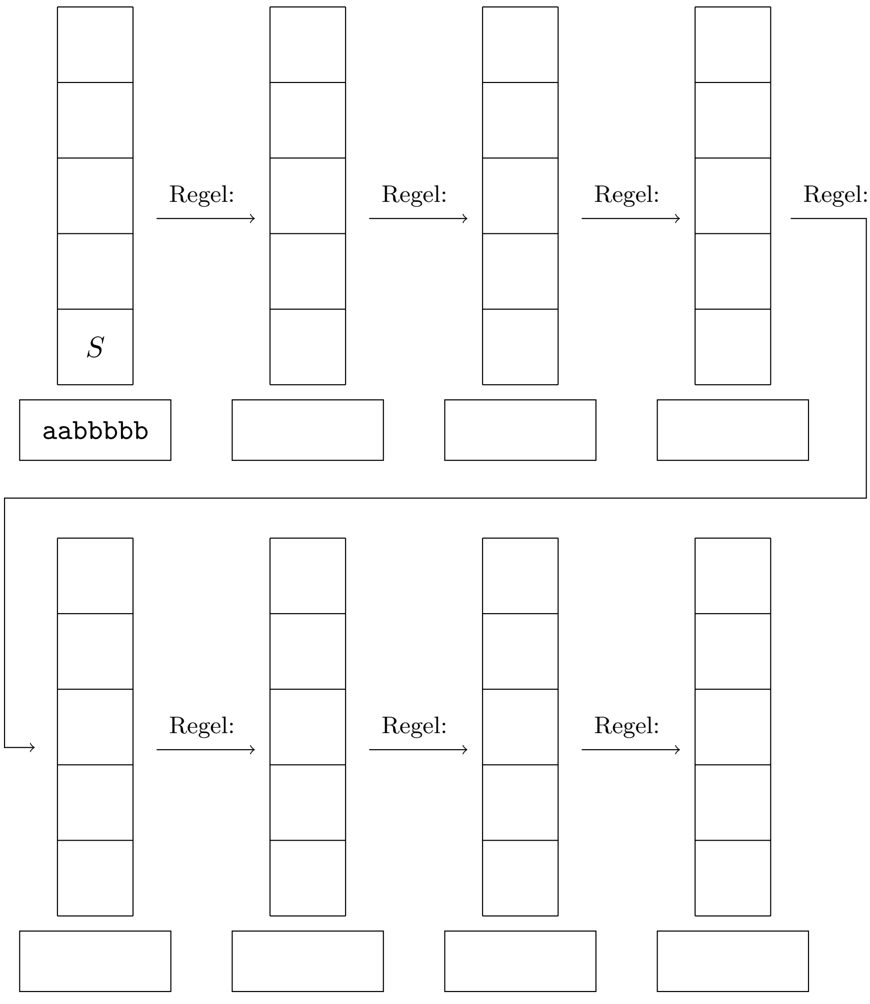
Wir geben zusätzlich noch eine Konstruktion zum Aufpumpen eines Wortes an. Da C nützlich ist, kommt es in einer Ableitung $S \xrightarrow{*} w$ mit $w \in \Sigma^*$ vor. Außerdem gelten $C \xrightarrow{*} abC$ und $C \xrightarrow{*} abb$. Damit sind alle Voraussetzungen zum Pumpen gemäß Pumping Lemma erfüllt und wir können mit dieser Konstruktion beliebig lange Wörter erzeugen.

Aufgabe 2 (3 Punkte)

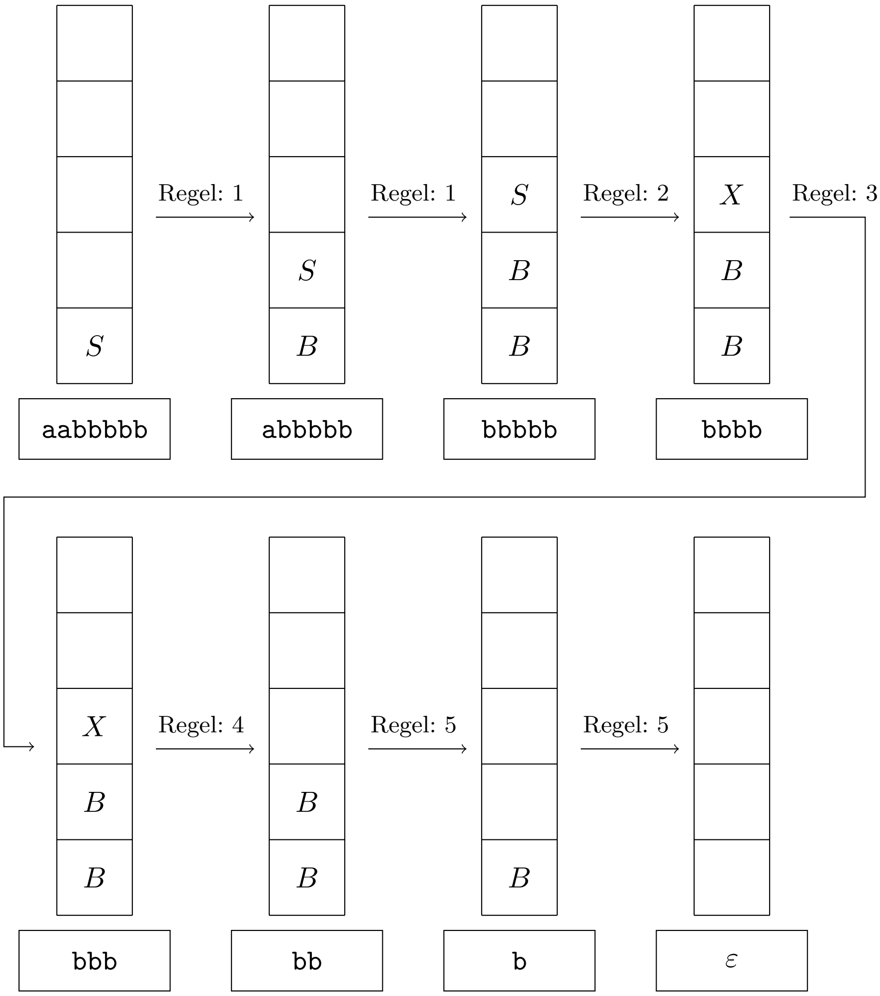
Die kontextfreie Grammatik G_3 in Greibach-Normalform über dem Eingabealphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ sei definiert durch die Menge der Nichtterminalsymbole $\{S, B, X\}$, Startsymbol S und folgenden Ableitungsregeln:

$$S \to \mathtt{a}SB \tag{1}$$
$$S \to \mathtt{b}X \tag{2}$$
$$X \to \mathtt{b}X \tag{3}$$
$$X \to \mathtt{b} \tag{4}$$
$$B \to \mathtt{b} \tag{5}$$

In Vorlesung und Übung wurde ein nichtdeterministischer Kellerautomat vorgestellt, der Sprachen erkennen kann, die von Grammatiken in Greibach-Normalform erzeugt werden. Geben Sie die Konfigurationen dieses Automaten an, die bei Abarbeitung der Eingabe `aabbbbb` entstehen. Vervollständigen Sie dazu folgendes Schema mit den zu lesenden Worten, den Stackinhalten und den Regeln, die beim Übergang verwendet werden. Ist das Eingabewort in der Sprache $L(G_3)$ enthalten?



Lösung:

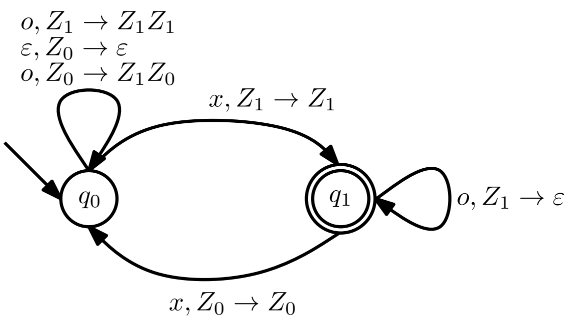


Aufgabe 3 (0,5 + 0,5 + 1 + 1 = 3 Punkte)

Betrachten Sie den Kellerautomaten

$$\mathcal{A} = (Q = \{q_0, q_1\}, \Sigma = \{o, x\}, \Gamma = \{Z_0, Z_1\}, q_0, Z_0, \delta, F = \{q_1\})$$

mit der Übergangsrelation δ gemäß dem folgenden Automatengraphen:



Hinweis: Beim Lesen von Zeichen x im Zustand q_1 und Z_1 auf dem STACK geht der Automat $\mathcal{A}$ in einen nicht-akzeptierenden Fehlerzustand.

(a) Welches in der Vorlesung vorgestellte Berechnungsmodell erweitert ein Kellerautomat? Inwiefern ist ein Kellerautomat eine Spezialisierung einer NTM?

(b) Ist $\mathcal{A}$ deterministisch? Begründen Sie Ihre Antwort!

(c) Geben Sie die Sprache L_F, die von $\mathcal{A}$ durch einen akzeptierenden Endzustand erkannt wird, an und begründen Sie Ihre Aussage. Ein formaler Beweis ist hierzu nicht nötig.

(d) Geben Sie die Sprache L_ε, die von $\mathcal{A}$ durch leeren Stack erkannt wird, an und begründen Sie Ihre Aussage. Ein formaler Beweis ist hierzu nicht nötig.

Lösung:

(a) Ein Kellerautomat ist eine Erweiterung eines NEA um einen unendlichen Speicher (auch: Keller oder Stack). Auf diesen Stack darf nur mittels FILO (First in last out) zugegriffen werden. Damit ist der Kellerautomat eine spezielle NTM mit 1-Weg-Eingabeband und speziellem Arbeitsband (Keller).

(b) Nein, der Automat $\mathcal{A}$ ist nicht deterministisch, da $|\delta(q_0, o, Z_0)| + |\delta(q_0, \varepsilon, Z_0)| > 1$.

(c) Im Zustand q_0 wird für jedes o ein Z_1 auf den Stack geschrieben und im Zustand q_1 wird für jedes o ein Z_1 vom Stack gelöscht. Der Kellerautomat wechselt beim Lesen von jedem x zwischen den Zuständen q_0 und q_1.

Suffix bei Beendung der Abarbeitung im Zustand q_1:
Betrachte die Konfigurationsfolge, nachdem das letzte Mal von q_1 nach q_0 gewechselt wurde. Wenn von q_1 nach q_0 gewechselt wird, geschieht dies mit dem Übergang $(q_0, Z_0) \in \delta(q_1, x, Z_0)$, es liegen also danach nur Z_0's auf dem Stack. Wenn nur Z_0 auf dem Stack liegt, dann wird nach q_1 gewechselt durch einlesen von $o^i x$ mit $i > 0$. Der Kellerautomat bleibt dann in Zustand q_1, solange höchstens i o's eingelesen werden, also wenn das Suffix des Wortes von der Form $o^i x o^j$ mit $i > 0$, $j \leq i$ ist.

Mögliche Schritte vor diesem Suffix:
Zuvor kann beliebig oft von q_0 nach q_1 und wieder zurück gewechselt werden. Von q_1 nach q_0 gewechselt werden kann nur, wenn gleich viele o's eingelesen werden wie zuvor im Zustand q_0. Insgesamt kann also beliebig oft ein Wort der Form $o^i x o^i x$ mit $i > 0$ eingelesen werden. Sei $L = \{o^i x o^i x \mid i > 0\}$, dann ist $L_F = L^* \cdot \{o^i x o^j \mid i > 0, j \leq i\}$.

(d) Der Stack kann nur geleert werden, wenn sich der Kellerautomat in Zustand q_0 befindet und keine Z_1 's auf dem Stack liegen. Wie bei Teilaufgabe (d) überlegt, ist dies genau am Anfang der Fall und wenn beliebig oft ein Wort der Form $o^i x o^i x$ mit $i > 0$ eingelesen wird. Sei $L = \{o^i x o^i x \mid i > 0\}$, dann ist L_ε also genau L^*.

Aufgabe 4 (2 Punkte)

Beweisen Sie, dass die Sprache

$$L_3 = \{w \in \{a, b, c\}^* \mid |w|_a = |w|_b = |w|_c\}$$

nicht kontextfrei ist, wobei $|w|_x$ die Häufigkeit des Zeichens x im Wort w bezeichne.

Lösung:
Wir nehmen an L_3 sei kontextfrei. Zu $n \in \mathbb{N}$ betrachten wir das Wort $w_n = a^{n+1} b^{n+1} c^{n+1} \in L_3$ und markieren in diesem Wort die $n + 1$ b's. Sei nun $w_n = uvwxy$ eine Zerlegung mit der Eigenschaft, dass von den n markierten Zeichen mindestens eines zu vx gehört und höchstens n zu vwx. Nach

Wahl der markierten Buchstaben in w_n enthält vwx entweder nur a's und b's oder nur b's und c's, da vwx ansonsten $n+1$ markierte Zeichen enthalten würde. Wir nehmen o.B.d.A. an, vwx enthalte nur a's und b's. Damit gilt $uv^0wx^0y = a^ib^jc^{n+1}$ mit $i < n+1$ oder $j < n+1$, da vx mindestens ein a oder ein b enthält, und somit $uv^0wx^0y \notin L_3$. Also erfüllt L_3 Ogden's Lemma nicht, was im Widerspruch zur Annahme steht. L_3 ist also nicht kontextfrei.

Aufgabe 5 (3 Punkte)

Zeigen oder widerlegen Sie, dass die kontextfreien Sprachen unter Spiegelung abgeschlossen sind. Die Spiegelung einer Sprache entsteht durch Spiegelung aller Wörter der Sprache, d.h. die Spiegelsprache L^R ist gegeben durch $L^R = \{w^R \mid w \in L\}$. $w^R = w_k \cdots w_1$ bezeichne dabei das Spiegelwort von $w = w_1 \cdots w_k$.

Lösung:
Die kontextfreien Grammatiken sind unter Spiegelung abgeschlossen. Sei $G = (V, \Sigma, S, R)$ eine kontextfreie Grammatik. Eine kontextfreie Grammatik $G^R = (V, \Sigma, S, R^R)$ für die Spiegelsprache lässt sich konstruieren, in dem man für jede Regel $A \to \beta$ aus R eine gespiegelte Regel $A \to \beta^R$ zu R^R hinzunimmt. Zu zeigen ist $L(G)^R = L(G^R)$.

Wir zeigen zunächst die folgende Beobachtung per Induktion über k: Sei $R_1, R_2, \ldots, R_k$ eine Folge von Regeln einer Ableitung $S \to \alpha$ mit $\alpha \in (V \cup \Sigma)^*$ mit $R_i \in R$. Dann ist $R_1^R, R_2^R, \ldots, R_k^R$ eine Folge von Regeln einer Ableitung $S \to \alpha^R$ mit $R_i^R \in R^R$.

Für $k = 1$ ist die Behauptung offensichtlich nach Konstruktion von R^R richtig.

Sei $R_1, R_2, \ldots, R_k$ eine Folge von Regeln einer Ableitung $S \xrightarrow{*} \alpha\beta\gamma$ mit $\alpha, \beta, \gamma \in (V \cup \Sigma)^*$ mit $R_i \in R$. Dann gilt $S \xrightarrow{*} \alpha A \gamma \xrightarrow{R_k} \alpha\beta\gamma$ mit $R_k : A \to \beta$. Nach Induktionsannahme liefern die Regeln $R_1^R, \ldots R_{k-1}^R$ die Ableitung $S \xrightarrow{*} (\alpha A \gamma)^R = \beta^R A \alpha^R$. Wegen $R_k^R : S \to \beta^R$ liefert die Anwendung von R_k^R auf $\gamma^R A \alpha^R$ dann die Ableitung $S \xrightarrow{*} \gamma^R \beta^R \alpha^R = (\alpha\beta\gamma)^R$.

Sei w also aus $L(G)$, dann gibt es eine Ableitung von w^R über der Regelmenge R^R, also $L(G)^R \subseteq L(G^R)$. Wegen der Dualität $(G^R)^R = G$ gilt obige Induktion auch für die gespiegelte Grammatik, woraus die Umkehrung analog folgt.

Aufgabe 6 (2 Punkte)

Die kontextfreie Grammatik G_1 über dem Eingabealphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ sei definiert durch die Menge der Nichtterminalsymbole $\{S, L, R\}$ mit dem Startsymbol S und folgenden Ableitungsregeln:

$$S \to LR \mid L \mid R$$
$$L \to \mathtt{a}L\mathtt{b} \mid \mathtt{ab}$$
$$R \to \mathtt{b}R\mathtt{a} \mid \mathtt{ba}$$

Geben Sie eine Grammatik G_2 in Greibach-Normalform an, sodass $L(G_2) = L(G_1)$ gilt.

Lösung:

$$S \to \mathtt{a}LBR \mid \mathtt{a}BR \mid \mathtt{a}LB \mid \mathtt{a}B \mid \mathtt{b}RA \mid \mathtt{b}A$$
$$L \to \mathtt{a}LB \mid \mathtt{a}B$$
$$R \to \mathtt{b}RA \mid \mathtt{b}A$$
$$A \to \mathtt{a}$$
$$B \to \mathtt{b}$$

Aufgabe 7 (5 + 1 = 6 Punkte)

In der Vorlesung haben Sie das Berechnungsmodell des Kellerautomaten kennengelernt. In dieser Aufgabe geht es darum, ein verwandtes Berechnungsmodell zu untersuchen. Ein *Schlangenautomat* entsteht aus einem Kellerautomaten, indem statt einem Kellerspeicher eine Warteschlange S verwendet wird. Die Warteschlange unterstützt zwei Operationen `push(`γ`)` und `pop`, die jeweils ein Zeichen γ *vorne* in S einfügen, beziehungsweise das *hinterste* Zeichen aus S entfernen.

(a) Wie mächtig sind deterministische Schlangenautomaten, d.h., welchen Typ von Sprachen können sie erkennen? Beweisen Sie Ihre Behauptung!

(b) Sind deterministische Schlangenautomaten gleich mächtig wie nichtdeterministische Schlangenautomaten? Begründen Sie!

Lösung:

(a) Deterministische Schlangenautomaten (DSA) können genau die Typ-0 Sprachen erkennen, denn sie sind gleich mächtig wie Turingmaschinen. Einerseits kann eine DTM natürlich einen DSA simulieren. Umgekehrt kann ein DSA auch eine DTM simulieren. Gehe dazu folgendermaßen vor. Sei L eine Sprache, die von einer DTM $\mathcal{M}$ erkannt wird. Wir gehen davon aus, dass alle Wörter in L mit einem speziellen Symbol $\Box$ enden[1]. Konstruiere nun einen DSA $\mathcal{A}$, der L erkennt. Als erstes liest $\mathcal{A}$ die komplette Eingabe. Das spezielle Endsymbol $\Box$ wird hier benötigt, damit $\mathcal{A}$ feststellen kann, wann das Wort w komplett eingelesen wurde, damit anschließend die Simulation von $\mathcal{M}$ beginnen kann. Jedes Eingabesymbol wird in die Schlange kopiert, bis schließlich das komplette Eingabewort w in der Schlange liegt. Dann beginnt die Simulation von $\mathcal{M}$ auf w. Zunächst wird das erste Symbol a in der Schlange durch ein neues Symbol (a, s) ersetzt, wobei s der Startzustand von $\mathcal{M}$ ist. Sei $\delta(a, s) = (b, q, d)$ mit $d \in \{L, R, N\}$. Falls $d = N$ ist ersetze (a, s) durch (b, q). Falls $d = R$ ist ersetze (a, s) durch b, entferne es aus Schlange und füge es am Ende der Schlange wieder ein. Ersetze dann das neue erste Symbol c durch (c, q). Falls $d = L$ ist muss das letzte Symbol der Schlange zum ersten Symbol der Schlange werden. Um das zu erreichen kann beispielsweise ein Markiersymbol in die Schlange eingefügt werden und eine Art Zwischenspeicher konstruiert werden, der immer das letzte Symbol, das in die Schlange eingefügt wurde, speichert.

(b) Ja, denn nichtdeterministische Turingmaschinen sind gleich mächtig wie deterministische Turingmaschinen, die Äquivalenz von Schlangenautomaten zu deterministischen Turingmaschinen lässt sich also einfach auf nichtdeterministische Turingmaschinen erweitern.

[1] Diese Einschränkung hätte in der Aufgabenstellung vorgegeben sein sollen.

Aufgabe 8 (1 + 1 + 1 = 3 Punkte)

Sei $\Sigma = \{a, b, c, d\}$ gegeben mit folgenden Häufigkeiten:

$$p(\sigma) = \begin{cases} 1/3 & \text{für } \sigma = a, b \\ 2/9 & \text{für } \sigma = c \\ 1/9 & \text{für } \sigma = d \end{cases}$$

(a) Benutzen Sie die Huffmancodierung, um einen optimalen präfixfreien Code zu erhalten.

(b) Benutzen Sie die Huffmancodierung, um einen anderen optimalen präfixfreien Code als den aus Teilaufgabe (a) zu erhalten.

(c) Geben Sie einen optimalen präfixfreien Code an, der keiner Huffmancodierung entspricht.

Lösung:

(a) $a \mapsto \texttt{0}$, $b \mapsto \texttt{10}$, $c \mapsto \texttt{110}$, $d \mapsto \texttt{111}$

(b) $a \mapsto \texttt{10}$, $b \mapsto \texttt{0}$, $c \mapsto \texttt{110}$, $d \mapsto \texttt{111}$

(c) $a \mapsto \texttt{00}$, $b \mapsto \texttt{10}$, $c \mapsto \texttt{01}$, $d \mapsto \texttt{11}$

Aufgabe 9 (1 + 1 + 3 = 5 Punkte)

Sei $\Sigma = \{a, b, c, d\}$ gegeben, wobei für die Häufigkeiten $p(a) > p(b) > p(c) > p(d)$ gilt.

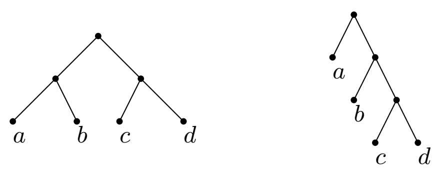
(a) Welche Codewortlängen $\{l_a, l_b, l_c, l_d\}$ sind für eine Huffmancodierung möglich?

(b) Angenommen es gilt $p(a) > p(b) + p(c)$. Was sind jetzt die möglichen Codewortlängen?

(c) Was ist der kleinste Wert ρ so dass aus $p(a) > \rho$ folgt, dass $l_a = 1$ ist? Beweisen Sie!

Lösung:

(a) Es gibt genau zwei nicht-isomorphe gewurzelte Binärbäume mit vier Blättern.



Die entsprechenden Codelängen sind $l_a = l_b = l_c = l_d = 2$ bzw. $l_a = 1, l_b = 2, l_c = l_d = 3$. Der erste Baum entsteht z.B. bei den Wahrscheinlichkeiten

$$p(a) = 0,31, \quad p(b) = 0,29, \quad p(c) = 0,21, \quad p(d) = 0,19.$$

Der zweite Baum entsteht z.B. bei den Wahrscheinlichkeiten

$$p(a) = 4/9, \quad p(b) = 3/9, \quad p(c) = 2/9, \quad p(d) = 1/9.$$

(b) Im ersten Schritt der Konstruktion des Huffmanbaums werden die beiden kleinsten Elemente c und d zusammengefasst. Aus $p(a) > p(b) + p(c)$ folgt direkt $p(a) > p(c) + p(d)$. Damit sind die nächsten beiden kleinsten Elemente b und der Teilbaum, der c und d enthält, zusammengefasst. Es bleibt also nur der rechte Baum möglich.

(c) Der kleinste Wert ist $\rho = 2/5$. Zeige, dass aus $p(a) > 2/5$ folgt, dass $l_a = 1$ ist. Nehme $l_a = 2$ an. Dann impliziert $p(c)+p(d) > p(a) > 2/5$ woraus $p(c) > 1/5$ folgt (wegen $p(c) > p(d)$) und $p(b) > 1/5$ (wegen $p(b) > p(c)$), aber $p(a)+p(c)+p(d) > p(a) > 4/5$, was $p(b) < 1/5$ impliziert (da die Summe aller Häufigkeiten Eins ergibt). Widerspruch. Also folgt aus $\rho > 2/5$, dass $l_a = 1$ ist. Ist dies das aber der kleinste Wert für ρ? Für jedes $2/5 > \epsilon > 0$ mit $p(a) = 2/5 - 6\epsilon$ finden wir Werte $p(b), p(c), p(d)$, so dass $p(a) < p(c) + p(d)$ gilt, woraus $l(a) = 2$ folgt, etwa $p(b) = 1/5 + 3\epsilon$, $p(c) = 1/5 + 2\epsilon$ und $p(d) = 1/5 + \epsilon$.