Karlsruher Institut für Technologie

Klausur Grundbegriffe der Informatik
WS 2021/2022

07. März 2022

Name:	
Matrikelnummer:	

Beachten Sie:

- Schreiben Sie Ihren vollständigen **Namen** und **Matrikelnummer** in Druckschrift in das Feld auf dem Deckblatt!
- Schreiben Sie Ihre Matrikelnummer oben auf jedes bearbeitete Aufgabenblatt.
- Schreiben Sie Ihre Lösungen auf die Aufgabenblätter. Bei Bedarf können Sie weiteres Papier anfordern.
- Wir akzeptieren auch englische Antworten.
- Sie haben **120 Minuten** Bearbeitungszeit.
- Die Klausur umfasst 19 Seiten (11 Blätter) mit 7 Aufgaben.

Aufgabe	1	2	3	4	5	6	7	**Gesamt**
Erreichte Punkte								
Erreichbare Punkte	9	10	7	6	10	6	12	**60**

Note

Aufgabe 1: Allgemeine Fragen (9 Punkte)

a) Begründen oder widerlegen Sie: für Mengen A, B und C gilt: **(1 Punkt)**

$$A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C) \ .$$

Musterlösung

Wir zeigen die Gleichheit durch zwei Implikationen:

"⇒" : Sei $x \in A \setminus (B \cup C)$. Dann ist $x \in A$ und $x \notin (B \cup C)$, also $x \notin B$ und $x \notin C$. Somit ist $x \in (A \setminus B)$ und $x \in (A \setminus C)$, also auch $x \in (A \setminus B) \cap (A \setminus C)$.

"⇐" : Sei $x \in (A \setminus B) \cap (A \setminus C)$. Dann ist $x \in (A \setminus B)$ und $x \in (A \setminus C)$. Also $x \in A$ und $x \notin B$ und $x \notin C$. Daher auch $x \in A \setminus (B \cup C)$.

b) Eine Funktion $f : \mathbb{N}_0 \to \mathbb{N}_0$ sei induktiv definiert durch $f(0) = 1$, $f(1) = 2$ und $f(n+1) = 2 \cdot f(n) + f(n-1)$ für $n \geq 1$. Geben Sie $f(2)$, $f(3)$, $f(4)$ an. Geben Sie außerdem $f(n+2)$ als Ausdruck unter Verwendung von $f(n)$ und $f(n-1)$ an. **(2 Punkte)**

$f(2) = \boxed{}$ $\quad f(3) = \boxed{}$ $\quad f(4) = \boxed{}$

$f(n+2) = \boxed{}$

Musterlösung

$f(2) = \boxed{5}$ $\quad f(3) = \boxed{12}$ $\quad f(4) = \boxed{29}$

$f(n+2) = \boxed{5f(n) + 2f(n-1)}$

c) Seien P, Q und R aussagenlogische Variablen. Geben Sie eine zur Formel

$$(P \to (P \to Q)) \to R$$

äquivalente Formel an, in der nur die Konnektive $\vee$, $\wedge$ und $\neg$ vorkommen, aber keine Implikation $\to$. Geben Sie eine vereinfachte Formel an, sodass jede Variable und jedes Konnektiv nur noch einmal vorkommt. **(1 Punkt)**

Name: MUSTERLÖSUNG Matrikelnummer: ____________

Musterlösung

$$
\begin{aligned}
& (P \to (P \to Q)) \to R \\
\equiv\ & (P \to (\neg P \lor Q)) \to R \\
\equiv\ & (\neg P \lor (\neg P \lor Q)) \to R \\
\equiv\ & (\neg P \lor Q) \to R \\
\equiv\ & \neg(\neg P \lor Q) \lor R \\
\equiv\ & (P \land \neg Q) \lor R
\end{aligned}
$$

d) Ordnen Sie die folgenden Funktionen so an, dass $f \in O(g)$ genau dann, wenn f links von g eingeordnet ist. **(2 Punkte)**

$$2^n, \quad 10^{100}, \quad 3^{n/2}, \quad n \log n, \quad n^5, \quad n \log\log n$$

Musterlösung

10^{100}	$n \log\log n$	$n \log n$	n^5	$3^{\frac{n}{2}}$	2^n

e) Geben Sie die Anzahl der Knoten und Kanten an, die der *ungerichtete* Graph mit der folgenden Adjazenzmatrix besitzt: **(1 Punkt)**

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Anzahl Knoten: ____

Anzahl Kanten: ____

Musterlösung

Anzahl Knoten: 5

Anzahl Kanten: 15

f) Beweisen oder widerlegen Sie: Wenn eine Relation $R \subseteq M \times M$ nicht symmetrisch ist, ist sie antisymmetrisch. **(2 Punkte)**

Musterlösung

Die Relation $R = \{(1,2),(2,1),(1,3)\}$ ist weder symmetrisch noch antisymmetrisch.
Nicht symmetrisch, da $(1,3) \in R$, aber $(3,1) \notin R$.
Nicht antisymmetrisch, da $(1,2),(2,1) \in R$ aber $1 \neq 2$.

Name: MUSTERLÖSUNG Matrikelnummer: ____________

Aufgabe 2: Homorphismen (10 Punkte)

Seien die Alphabete $A = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ und $B = \{\mathtt{0}, \mathtt{1}\}$ gegeben. Sei ferner $h : A^* \to B^*$ ein Homomorphismus mit $h(\mathtt{a}) = \mathtt{01}$, $h(\mathtt{b}) = \mathtt{010}$ und $h(\mathtt{c}) = \mathtt{1}$.

a) Geben Sie $h(\mathtt{abc})$ und $h(\mathtt{caa})$ an. **(1 Punkt)**

$h(\mathtt{abc}) =$

$h(\mathtt{caa}) =$

Musterlösung

$\mathrm{h}(\mathtt{abc}) = \mathtt{010101}$
$\mathrm{h}(\mathtt{caa}) = \mathtt{10101}$

b) Zeigen Sie, dass h nicht injektiv ist. **(1 Punkt)**

Musterlösung

Für $w_1 = \mathtt{aa}$ und $w_2 = \mathtt{bc}$ ist $h(w_1) = h(w_2) = \mathtt{0101}$. Daher ist h nicht injektiv.

c) Wir wollen nun eingehender untersuchen, für welche Wortmengen $\{u, w\}$ mit $u \neq w$ die Eigenschaft $h(u) = h(w)$ gilt. Geben Sie in einem ersten Schritt zwei notwendige Kriterien basierend auf Wortlängen oder Anzahl der vorkommenden Zeichen an, damit diese Eigenschaft gilt. In den von Ihnen angegebenen Kriterien soll h nicht explizit vorkommen. **(2 Punkte)**

Musterlösung

Z.B.:

- Anzahl Zeichen $\mathtt{0}$: $N_{\mathtt{a}}(u) + 2 \cdot N_{\mathtt{b}}(u) = N_{\mathtt{a}}(w) + 2 \cdot N_{\mathtt{b}}(w)$
- Anzahl Zeichen $\mathtt{1}$: $N_{\mathtt{a}}(u) + N_{\mathtt{b}}(u) + N_{\mathtt{c}}(u) = N_{\mathtt{a}}(w) + N_{\mathtt{b}}(w) + N_{\mathtt{c}}(w)$
- Wortlänge: $2 \cdot N_{\mathtt{a}}(u) + 3 \cdot N_{\mathtt{b}}(u) + N_{\mathtt{c}}(u) = 2 \cdot N_{\mathtt{a}}(w) + 3 \cdot N_{\mathtt{b}}(w) + N_{\mathtt{c}}(w)$
- Wortlänge: $|u| = |w|$ (wegen Anzahl Zeichen $\mathtt{1}$)

d) Geben Sie nun alle Wortmengen $\{u, w\}$ mit $u \neq w$, $|u| \leq 4$ und $|w| \leq 4$ an, für die $h(u) = h(w)$ gilt. **(3 Punkte)**

Musterlösung

Aus dem zweiten Kriterium von c) folgt, dass $|u| = |w|$ gelten muss für $h(u) = h(w)$, daher können wir nach Wortlängen unterscheiden: (für Längen 0 und 1 gibt es keine Lösungen)

- Länge 2: $\{\mathtt{aa}, \mathtt{bc}\}$ (1 Lsg.)
- Länge 3:
  $\{xaay, xbcy\}$ für alle $x, y \in A^*, |xy| = 1$ und $\{abc, bca\}$ (7 Lsg.)
- Länge 4:
  $\{xaay, xbcy\}$ für alle $x, y \in A^*, |xy| = 2$ (27 Lsg.) und
  $\{xabcy, xbcay\}$ für alle $x, y \in A^*, |xy| = 1$ (6 Lsg.) und
  $\{aaaa, bcbc\}, \{aabc, bcaa\}, \{abca, bcbc\}$ (3 Lsg.)

Allgemeine Lösung (für beliebige Wortlängen): Alles, was sich durch (u.U. mehrfache parallele) Ersetzungen anhand der folgenden beiden Gleichungen ergibt:

$$\mathtt{a}(\mathtt{bc})^i\mathtt{a} = \mathtt{b}(\mathtt{cb})^i\mathtt{c}$$
$$\mathtt{a}(\mathtt{bc})^i\mathtt{bc} = \mathtt{b}(\mathtt{cb})^i\mathtt{ca}$$

e) Durch den Homomorphismus h wird eine Äquivalenzrelation $\equiv_h$ auf A^* induziert:

$$\forall u, w \in A^* \;:\; u \equiv_h w \;\leftrightarrow\; h(u) = h(w) \;.$$

Wir betrachten nun die Faktormenge $A^*/_{\equiv_h}$ (d.h. die Menge aller Äquivalenzklassen). Zeigen Sie: **(3 Punkte)**
(1) Es gibt unendlich viele verschiedene Äquivalenzklassen $[w]_{\equiv_h} \in A^*/_{\equiv_h}$.
(2) Es gibt keine Äquivalenzklasse mit unendlich vielen Elementen.

Musterlösung

Ad (1): Betrachte die Wortmenge $\{\mathtt{c}^i \mid i \in \mathbb{N}_+\}$. Für zwei beliebige verschiedene Worte w_1, w_2 aus dieser Menge gilt $h(w_1) \neq h(w_2)$, da die Anzahl der `1`en sich unterscheiden. Also gibt es unendlich viele Äquivalenzklassen.

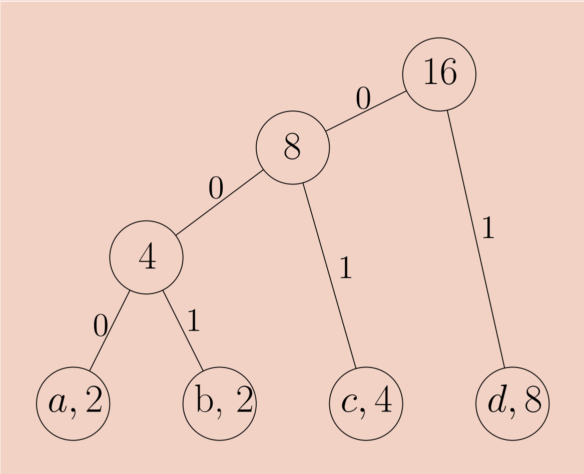
Ad (2): Alle äquivalenten Worte müssen dieselbe Länge haben. Über dem 3-elementigen Alphabet A kann es aber nur 3^n viele Worte der Länge n geben, also endlich viele.

Name: MUSTERLÖSUNG Matrikelnummer: ____________

Aufgabe 3: Huffman-Codierung (7 Punkte)

a) Erstellen Sie einen Huffman-Baum für das Wort $w = \texttt{b}^2\texttt{d}^4\texttt{c}^4\texttt{a}^2\texttt{d}^4$ über dem Alphabet $A = \{\texttt{a}, \texttt{b}, \texttt{c}, \texttt{d}\}$ und geben Sie die daraus resultierende Codierung von w an. **(2 Punkte)**

Musterlösung



Daraus ergibt sich die Codierung $H(w) = (001)^2 1^4 (01)^4 (000)^3 1^4$

b) Wir wollen nun ein Wort w über einem beliebigen Alphabet A mit mindestens zwei Symbolen betrachten, dessen Länge eine Zweierpotenz ist, d.h. $|w| = n = 2^k$ für ein $k \in \mathbb{N}_+$, und zu dem wir das Folgende wissen: Jedes Zeichen $a \in A$ kommt in dem Wort mit der Häufigkeit einer Zweierpotenz vor, d.h., es gilt für alle $a \in A$ gibt es ein $j_a \in \mathbb{N}_0$ mit $N_a(w) = 2^{j_a}$. In dieser Aufgabe soll gezeigt werden, dass unter diesen Annahmen die Huffman-Codierung $H(a)$ eines Zeichens $a \in A$ mit Häufigkeit 2^{j_a} genau Länge $k - j_a$ hat. **(5 Punkte)**

Hinweis: Unter den gemachten Annahmen gibt es mindestens zwei Zeichen gleicher geringster Häufigkeit in w. Sie dürfen die Korrektheit dieser Aussage in Ihrem Beweis annehmen.

i) Beweisen Sie die Aussage für ein Alphabet A mit zwei Zeichen, d.h. $|A| = 2$. (1 Punkt)

Musterlösung

O.B.d.A. sei $A = \{a, b\}$ und $|w| = n = 2^k$. Da a und b die einzigen Zeichen in A sind, gilt $|w| = N_a(w) + N_b(w) = 2^k$ und nach dem *Hinweis* ist $N_a(w) = N_b(w)$. Es folgt

$$N_a(w) + N_b(w) = 2^k \quad (1)$$
$$(\text{Hinweis}) \Rightarrow 2 \cdot N_a(w) = 2^k \quad (2)$$
$$\Leftrightarrow 2 \cdot 2^{j_a} = 2^k \quad (3)$$
$$\Leftrightarrow 2^{j_a} = 2^{k-1} \quad (4)$$
$$\Rightarrow j_a = k - 1 \quad (5)$$

Da der Huffman-Baum für ein Alphabet mit zwei Zeichen aus einer Wurzel besteht, die jeweils direkt über eine Kante mit den Blattknoten verbunden ist, gilt $|H(a)| = 1 = k - j_a = k - (k - 1)$.

ii) Beweisen Sie die Aussage für den allgemeinen Fall. (4 Punkte)

Hinweis: Eine Möglichkeit des Beweises ist per Induktion über die Mächtigkeit $m = |A|$ des Alphabets.

Musterlösung

Induktionsanfang ($m = 2$, da in Aufgabe $m > 1$ gefordert)
Wurde schon in Aufgabe (i) gezeigt.
Induktionsvoraussetzung Für ein beliebiges aber festes $m \in \mathbb{N}^+$, $m > 1$ gelte: $\forall a_i \in A, |A| = m : N_{a_i}(w) = 2^{j_{a_i}} \to |H(a_i)| = k - j_{a_i}$.
Induktionsschluss Sei $A = \{a_1, a_2, ..., a_n, a_{m+1}\}$ und $N_{a_i}(w)$ wie gefordert für alle a_i. Die $a_i \in A$ seien o.B.d.A. nach aufsteigender Häufigkeit indiziert (also $N_{a_i}(w) \leq N_{a_{i+1}}(w)$).
Nach Konstruktion des Huffman-Baumes werden zuerst a_1 und a_2 zu einem neuen Knoten $x_{1,2}$ mit $N_{x_{1,2}}(w) = N_{a_1}(w) + N_{a_2}(w)$ zusammengefasst. Da a_1, a_2 die *beiden* Zeichen mit *geringster Häufigkeit* $N_{a_1}(w) = N_{a_2}(w) = 2^{j_{a_1}}$ sind, gilt: $N_{x_{1,2}}(w) = 2 \cdot 2^{j_{a_1}} = 2^{j_{a_1}+1}$.
Konstruiere neues Alphabet $A' = \{x_{1,2}, a_3, a_4, ..., a_{m+1}\}$ mit $|A'| = |A| - 2 + 1 = m$ (a_1, a_2 entfernt, $x_{1,2}$ hinzugefügt). Die $a_3, ..., a_{m+1}$ haben immer noch Häufigkeiten der Form 2^{a_j} genauso wie $x_{1,2}$.
Mit der Induktionsvoraussetzung gilt dann die Behauptung für alle Zeichen in A'. Da $a_3, ..., a_{m+1}$ nicht verändert wurden, gilt die Behauptung für sie sofort. Die Länge der Huffman-Codierung $H(x_{1,2})$ ist dann ebenfalls nach Induktionsvoraussetzung $|H(x_{1,2})| = k - (j_{a_1} + 1)$. Nach Konstruktion der Huffman-Codierung ist der Knoten $x_{1,2}$ aus der Zusammenführung der Knoten für a_1 und a_2 entstanden. Damit ist $|H(a_i)| = |H(x_{1,2})| + 1 = k - j_{a_1}$, $i \in \{1, 2\}$ und die Länge aller Huffman-Codierungen ist wie in der Behauptung gefordert.

Name: MUSTERLÖSUNG Matrikelnummer: ____________

Aufgabe 4: Logik (6 Punkte)

a) Gegeben seien die drei aussagenlogischen Variablen P, Q und R. **(2 Punkte)**

i) Geben sie eine aussagenlogische Formel an, welche eine Tautologie ist und P und Q enthält.

Musterlösung

i) mehrere richtige Lösungen:
$P \vee \neg P \wedge Q \vee \neg Q$
$P \vee \neg P \vee Q \vee \neg Q$
$P \rightarrow P \wedge Q \rightarrow Q$
$P \rightarrow P \vee Q \rightarrow Q$

ii) Ist die aussagenlogische Formel $(P \rightarrow Q) \vee (Q \rightarrow R)$ eine Tautologie? Begründen Sie Ihre Antwort.

Musterlösung

Ja. Für $I(Q) = w$ ist $val_I(P \rightarrow Q) = w$ für beliebiges $I(P)$. Für $I(Q) = f$ ist $val_I(Q \rightarrow R) = w$ für $I(R)$ beliebig. Da beide Teilformeln mit einem $\vee$ (oder) verbunden sind und (mindestens) eine Teilformel immer wahr ist, gilt F ist eine Tautologie.

b) Die Studierenden Arthur, Bertha und Celina gehen manchmal in die Mensa um ihren Hunger zu stillen.

(a) Die Variable A bedeutet, Arthur ist in der Mensa.

(b) Die Variable B bedeutet, Bertha ist in der Mensa.

(c) Die Variable C bedeutet, Celina ist in der Mensa.

Drücken Sie folgende Aussagen als aussagenlogische Formeln mit den Variablen A, B und C aus. **(2 Punkte)**

i) Es sind genau zwei der drei Studierenden in der Mensa.

Musterlösung

a) $(A \wedge B \wedge \neg C) \vee (A \wedge \neg B \wedge C) \vee (\neg A \wedge B \wedge C)$

ii) Arthur ist nur in der Mensa, wenn Bertha und Celina nicht in der Mensa sind.

Musterlösung

$(\neg A) \vee (\neg B \wedge \neg C)$ oder
$A \rightarrow (\neg B \wedge \neg C)$

c) Ist die nachfolgende Formel erfüllbar? Begründen Sie Ihre Antwort entweder durch eine Wahrheitstabelle, Umformung oder durch ein Modell. **(2 Punkte)**

$$(A \rightarrow B) \land (B \rightarrow C) \land (C \rightarrow A) \land (A \lor B \lor C) \land (\neg A \lor \neg B \lor C)$$

Musterlösung

Erfüllbar mit Model $val_I(A) = w$, $val_I(B) = w$, $val_I(C) = w$

Name: MUSTERLÖSUNG Matrikelnummer: ____________

Aufgabe 5: Grammatiken (10 Punkte)

Gegeben seien die formalen Sprachen L_1, L_2 über dem Alphabet $\{a, b, c\}$:

$$L_1 = \{a^n b^m c^n \in \{a, b, c\}^* \mid n, m \in \mathbb{N}_0\}$$

$$L_2 = \{w \in \{a, b, c\}^* \mid (N_a(w) + N_b(w)) \bmod 3 = 0\}$$

a) Geben Sie die kleinste Anzahl an Nichtterminalsymbolen an, welche eine kontextfreie Grammatik G_1 mit $L(G_1) = L_1$ enthalten muss. Begründen Sie. **(1 Punkt)**

Musterlösung

G_1 benötigt (mindestens) 2 Nichtterminalsymbole.
Begründung: n und m sind unabhängig voneinander. Also muss P mindestens zwei Produktionen beinhalten, in deren rechten Seiten keine Terminalsymbole außer b bzw. außer a und c vorkommen. Diese beiden Produktionen können nicht das gleiche Nichtterminalsymbol auf der linken Seite teilen, denn sonst könnten Worte erzeugt werden, in denen sich a's / c's und b's sich wiederholt abwechseln oder die mit b's beginnen, aber trotzdem a's / c's enthalten.

b) Gibt es einen regulären Ausdruck R so, dass $L_2 = \langle R \rangle$? Falls ja, geben Sie R an. Falls nicht, begründen Sie kurz, warum das nicht der Fall sein kann. **(1 Punkt)**

Musterlösung

Ja: `c* ((a|b) c* (a|b) c* (a|b) c*)*`

c) Geben Sie eine kontextfreie Grammatik G_2 so an, dass $L(G_2) = L_1 \cap L_2$. Verwenden Sie höchstens 4 Nichtterminalsymbole. **(3 Punkte)**

Musterlösung

$G = (N, T, S, P)$ mit

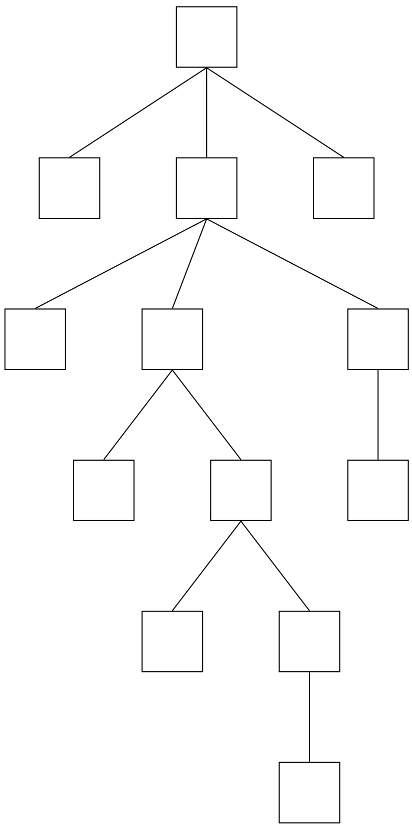
$$\begin{aligned}
N =& \{S, A_1, A_2, B\} \\
T =& \{a, b, c\} \\
P =& \{S \to aA_1c \mid B \\
& \quad A_1 \to aA_2c \mid bbB \\
& \quad A_2 \to aSc \mid bB \\
& \quad B \to bbbB \mid \varepsilon \\
& \}
\end{aligned}$$

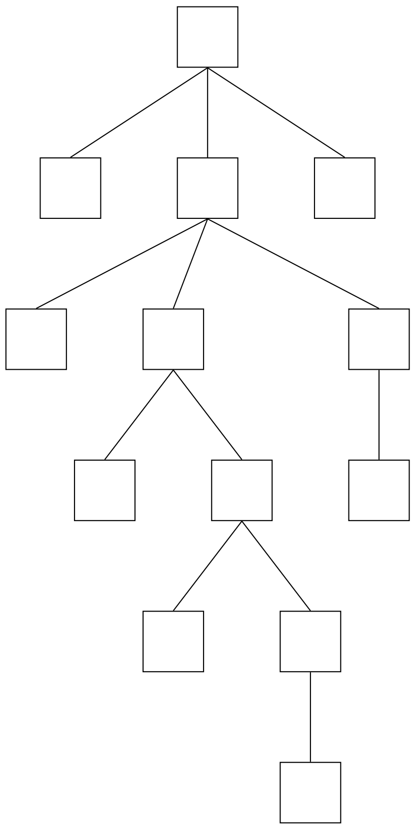
d) Betrachten Sie die kontextfreie Grammatik $G = (N, T, S, P)$ mit

$$
\begin{aligned}
N =& \{S, A, B, C, R\} \\
T =& \{a, b, c\} \\
P =& \{S \to aSa \mid bSb \mid cSc \mid aRA \mid bRB \mid cRC \\
& \quad R \to aR \mid bR \mid cR \mid \varepsilon \\
& \quad A \to b \mid c \\
& \quad B \to a \mid c \\
& \quad C \to a \mid b \\
& \}
\end{aligned}
$$

i) Unten finden Sie die Struktur eines Ableitungsbaumes. Füllen Sie diese so aus, dass sich ein gültiger Ableitungsbaum zu einem Wort aus $L(G)$ ergibt. Geben Sie das von Ihnen gewählte Wort an. **(2 Punkte)**

Wort: ____________________





Hinweis: Sie können den zweiten Baum für Nebenrechnungen verwenden. Wenn Sie beide Bäume verwenden, machen Sie unbedingt deutlich, welcher Baum Ihre Lösung enthält und gewertet werden soll.

Name: MUSTERLÖSUNG Matrikelnummer: ____________

ii) Geben Sie $L(G)$ explizit als Mengenausdruck an. **(1 Punkt)**

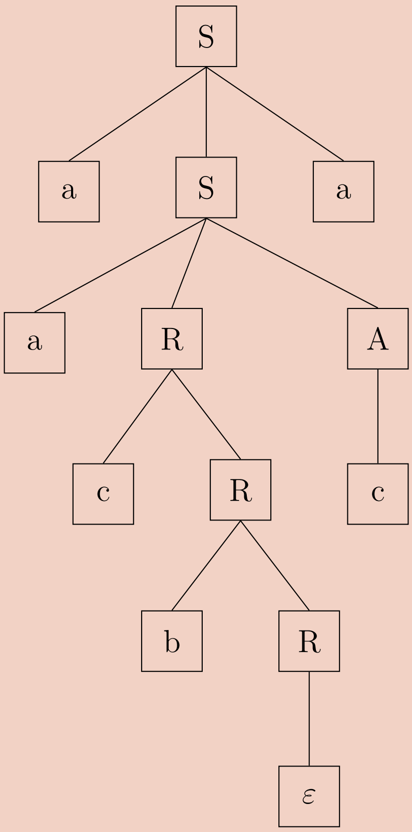
iii) Geben Sie eine bijektive Funktion $f : \{a,b,c\}^+ \rightarrow \{a,b,c\}^+$ an, sodass gilt:

$$\forall w \in \{a,b,c\}^+ : f(w) \in L(G) \Leftrightarrow w \in L(G) \enspace .$$

f darf dabei nicht die Identitätsfunktion sein. Begründen Sie die Korrektheit Ihrer Funktion. **(2 Punkte)**

Musterlösung

i) Bsp für das Wort $aacbca \in L(G)$:



ii)

$$L(G) = \{w \in \{a,b,c\}^+ \mid w \text{ ist kein Palindrom } \}$$

bzw.

$$L(G) = \{w \in a,b,c^+ | w = vxw', |v| = |w'|, w' \neq v^R, \\ v, w' \in \{a,b,c\}^+, x \in \{a,b,c,\epsilon\}\}$$

iii) $f : \{a,b,c\}^+ \rightarrow \{a,b,c\}^+, \quad f(w) = w^R$, wobei w^R für ein Wort $w \in \{a,b,c\}^+$ das Spiegelwort von w ist, leistet das Gewünschte.
Das Spiegelwort eines Wortes w ist genau dann ein Palindrom, wenn das Wort selbst ein Palindrom ist. Somit bildet f jedes Palindrom auf ein anderes Palindrom und jedes Nicht-Palindrom auf ein Nicht-Palindrom ab.

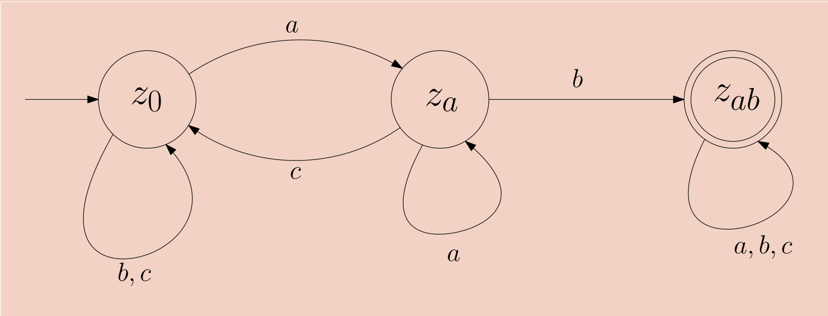
Aufgabe 6: Automaten und reguläre Ausdrücke (6 Punkte)

Gegeben sei das Alphabet $\Sigma = \{a, b, c\}$ und die beiden regulären Sprachen

- $L_1 = \{w \in \Sigma^* \mid ab \text{ kommt in } w \text{ vor}\}$
- $L_2 = \{w \in \Sigma^* \mid ab \text{ kommt nicht in } w \text{ vor}\}$

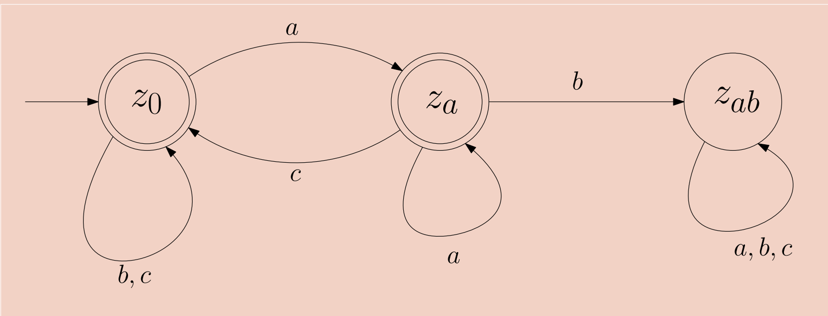
a) Geben Sie einen endlichen Akzeptor $A = (Z, z_0, f, F)$ mit $|Z| \leq 4$ an, der L_1 akzeptiert. **(1 Punkt)**

Musterlösung



b) Geben Sie einen endlichen Akzeptor $A' = (Z', z_0', f', F')$ mit $|Z'| \leq 4$ an, der L_2 akzeptiert. **(1 Punkt)**

Musterlösung



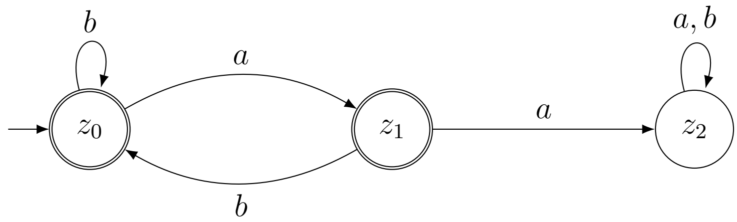
c) Geben Sie einen regulären Ausdruck an, der L_1 beschreibt. **(1 Punkt)**

Musterlösung

Ein gültiger regulärer Ausdruck wäre zum Beispiel $(a|b|c) * ab(a|b|c)*$.

Name: MUSTERLÖSUNG Matrikelnummer: ______________

d) Gegeben sei der unten aufgeführte Automat über dem Eingabealphabet $\{a, b\}$. Geben Sie einen regulären Ausdruck an, welcher die vom Automaten akzeptierte Sprache beschreibt. **(3 Punkte)**



Musterlösung

$(b^*|(ab)^*(a|\emptyset^*)$

Aufgabe 7: Turingmaschinen (12 Punkte)

a) Entwerfen Sie einen TM-Aktzeptor A mit Bandalphabet $X = \{a, b, c, \$, \square\}$, der die Sprache $L_{abc} = \{a^n b^n c^n \mid n \in \mathbb{N}_+\}$ akzeptiert. **(5 Punkte)**

i) Geben Sie zuerst in 1-2 Sätzen eine Idee für die Funktionsweise einer solchen Turingmaschine an. (1 Punkt)

Musterlösung

Das Eingabewort von links nach rechts durchgehen und nacheinander Zeichen a, b, c durch \$ ersetzen. Danach zurück an den Anfang gehen und den Prozess wiederholen und akzeptieren sobald nur noch $\$'s$ auf dem Band stehen.

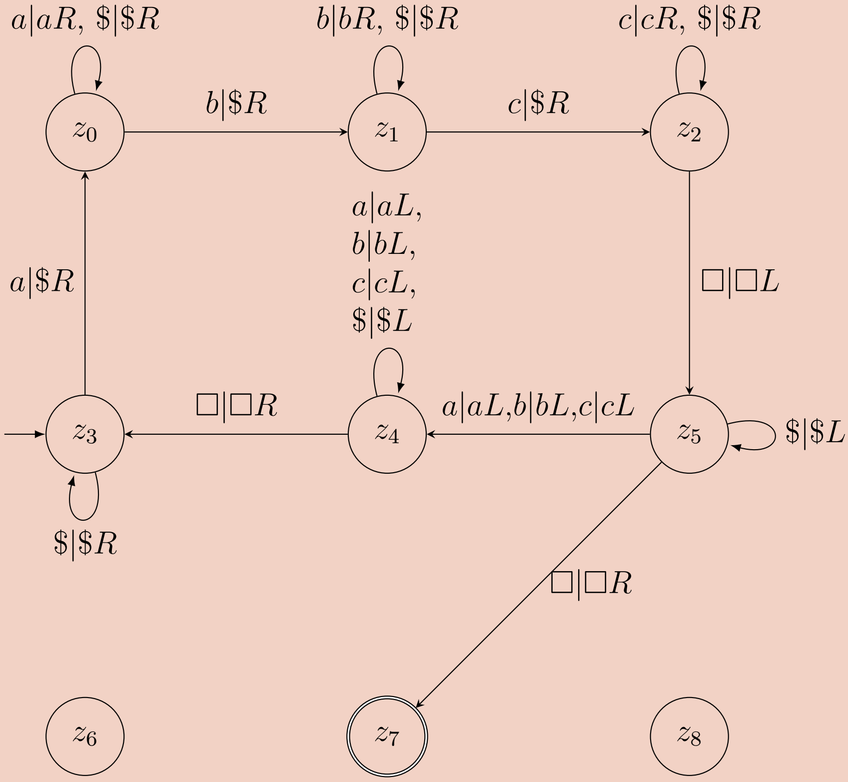
ii) Geben Sie nun eine Turingmaschine an. Nutzen Sie hierfür die vorgegebenen Zustände. Sie müssen nicht alle Zustände nutzen. Verwenden Sie die Notationen aus der Vorlesung und markieren Sie akzeptierende Zustände. (4 Punkte)

Name: MUSTERLÖSUNG Matrikelnummer: ____________________

Sie können die unten angegebene Zustandsmenge als Alternativlösung verwenden. Falls Sie dies tun, kennzeichnen Sie eindeutig, welche Lösung gewertet werden soll!

Musterlösung

Eine mögliche Lösung:



b) Wie müssten Sie Ihre Turingmaschine aus ii) verändern, wenn das Eingabealphabet um einen weiteren Buchstaben d erweitert wird und die Sprache

$$L_{abc}^{d} = \{d^v\ a^n\ d^x\ b^n\ d^y\ c^n\ d^z \mid n \in \mathbb{N}_0 \wedge v, x, y, z \in \mathbb{N}_0\}$$

akzeptiert werden soll. Beschreiben und begründen Sie Ihr Vorgehen in wenigen Sätzen. Zeichnen Sie keine neue Turingmaschine. **(1 Punkt)**

Musterlösung

- Anzahl d's ist beliebig und kann daher durch \$ ersetzt werden.
- Die d's dürfen aber nicht überall entfernt werden (Beispiel: adabbcc). Sie dürfen nur zwischen den anderen Buchstaben vorkommen, daher müssen neue Zustände eingefügt werden.
- Weiterhin muss das leere Wort akzeptiert werden, da jetzt $n \in \mathbb{N}_0$.

c) Sind die folgenden Aussagen bzgl. Turingmaschinen wahr oder falsch? Begründen Sie Ihre Antwort jeweils kurz. Ohne richtige Begründungen gibt es keine Punkte. **(2 Punkte)**

i) Für jeden TM-Akzeptor A gibt es einen TM-Akzeptor A' mit $L(A') = X^* \setminus L(A)$. (1 Punkt)

Musterlösung

FALSCH, da Turingmaschinen Wörter akzeptieren, wenn sie in einem akzeptierenden Zustand **halten**. Daher kann man nicht einfach die akzeptierenden Zustände umkehren, wie bei Automaten. Beispiel: Halteproblem.

ii) Jede Turingmaschine, welche durch ein endlich langes Codewort beschrieben werden kann, kann auch nur endlich viele Wörter akzeptieren. (1 Punkt)

Musterlösung

FALSCH, da jede Turingmaschinen durch ein endlich langes Codewort dagestellt werden kann. Zustände, Zeichen auf einem Arbeitsband und auch alle anderen Entitäten sind endlich.

Name: MUSTERLÖSUNG Matrikelnummer: ____________

d) Eine Rechts-Turingmaschine T_r kann auf dem Arbeitsband in jedem Schritt nur nach rechts gehen, also nicht stehen bleiben oder nach links gehen. Die partielle Bewegungsfunktion m_r ist also definiert durch $m_r(z, x) = R$ für alle $x \in X$ und alle $z \in Z$. Weiterhin ist die Zustandsübergangsfunktion für das $\square$-Symbol bei Turingmaschinen T_r nicht definiert. Ein Rechts-TM-Akzeptor ist ein Akzeptor basierend auf einer Rechts-Turingmaschine.

Zeigen Sie, dass Rechts-TM-Akzeptoren genauso mächtig sind wie endliche Akzeptoren. Genauso mächtig heißt hier, dass Sie die gleiche Menge an Sprachen entscheiden können. **(4 Punkte)**

Musterlösung

Rechts-TM-Akzeptoren und endliche Akzeptoren sind gleich mächtig. Wir zeigen dies durch Transformationen in beide Richtungen.

endlicher Akzeptor $\implies$ Rechts-TM-Akzeptor: (wurde in der Übung für normale Turingmaschinen gezeigt)

- Zustände und akzeptierende Zustände sind bei Rechts-TM die gleichen wie beim Automaten
- Die Eingabe liegt von links nach rechts auf dem Arbeitsband
- Zustandsübergänge bleiben gleich.
- Es wird immer das Zeichen geschrieben welches gelesen wird (ist unwichtig, da das Zeichen nie wieder eingelesen wird, da die Rechts-TM auf dem Arbeitsband immer nur nach rechts geht).
- Bei jedem Schritt geht man einen Schritt nach rechts.

Rechts-TM-Akzeptor $\implies$ endlicher Akzeptor :

- Zustände und akzeptierende Zustände sind beim Automaten die gleichen wie bei der Rechts-TM
- Das Arbeitsband entspricht der Eingabe eines Automaten, da nur von links nach rechts einmal über das Arbeitsband gelaufen werden kann.
- Zustandübergänge:

  Die definierten Zustandübergänge bleiben gleich. Das geschriebene Zeichen der rechts TM kann im Automaten "gelöscht"werden, da es nie wieder eingelesen wird und damit die Funktionisweise der Turingmaschine nicht beeinflusst. Für undefinierte Zustandsübergänge müssen zwei neue Zustände eingefügt werden. Wenn die Turingmaschine in einem akzeptierenden Zustand ist und ein undefinierter Zustandsübergang stattfindet, geht der Automat in den neuen akzeptierenden Zustand. Wenn die Turingmaschine in einem nicht akzeptierenden Zustand ist, geht der Automat in einen nicht akzeptierenden Zustand. Der Automat bleibt für alle Eingaben in dem Zustand bis die Eingabe fertig abgearbeitet ist.