Lösungsvorschläge und Erläuterungen

Klausur zur Vorlesung
Grundbegriffe der Informatik
10. September 2019

Nachname:							
Vorname:							
Matr.-Nr.:							

Diese Klausur ist mein ☐ 1. Versuch ☐ 2. Versuch in GBI

Falls 2. Versuch, bitte *gut lesbar* ausfüllen:

Email-Adr.:
Postanschrift:

Aufgabe	1	2	3	4	5	6	7
max. Punkte	7	8	6	7	6	6	5
tats. Punkte							

Gesamtpunktzahl:	/ 45		Note:	

/ 7

Aufgabe 1 (2 + 1 + 2 + 2 = 7 Punkte)

/ 2

a) Für $i \in \{1,2,3,4,5\}$ seien Funktionen $f_i : \mathbb{N}_+ \to \mathbb{N}_+$ wie folgt definiert:

- $f_1(n) = 2^{(n^n)}$
- $f_2(n) = (2^n)^n$
- $f_3(n) = n^{(2^n)}$
- $f_4(n) = (n^2)^n$
- $f_5(n) = (n^n)^2$

Geben Sie die Zahlen in $\{1,2,3,4,5\}$ in einer Reihenfolge x_1, x_2, x_3, x_4, x_5 so an, dass für alle $1 \le i < 5$ gilt: $f_{x_i} \in O(f_{x_{i+1}})$.

$x_1 =$ 4 $\quad x_2 =$ 5 $\quad x_3 =$ 2 $\quad x_4 =$ 3 $\quad x_5 =$ 1

Erläuterungen:

- f_2 vor f_1, weil $f_2(n) = 2^{(n^2)}$ einen schwächer wachsenden Exponenten hat als f_1.
- $f_4(n) = f_5(n) = n^{2n}$, also Reihenfolge egal, aber $f_3(n)$ danach, weil der Exponent 2^n stärker wächst als der Exponent $2n$.
- $f_3(n) = 2^{(2^n \log n)} = 2^{(2^n 2^{\log\log n})} = 2^{(2^{n+\log\log n})}$ noch vor $f_1 = 2^{(2^{n \log n})}$
- analog $f_4(n) = n^{2n} = 2^{2n \log n}$ noch vor $f_2(n) = 2^{(n^2)}$
- analog $f_2(n) = 2^{(n^2)}$ noch vor $f_3(n) = 2^{(2^n \log n)}$

/ 1

b) Gegeben ist die Grammatik $G = (N, T, X, P)$ mit $N = \{X, Y\}$, $T = \{a, b\}$ und $P = \{X \to aYb, Y \to aY, Y \to \varepsilon\}$. Geben Sie Produktionen p_1 und p_2 an, sodass die Grammatik $G' = (N, T, X, P')$ mit $P' = P \cup \{p_1, p_2\}$ die Sprache $L(G') = \{a\}^* \cdot \{b\}^*$ erzeugt:

$p_1 =$ $X \to Y$ $\qquad p_2 =$ $Y \to Yb$

/ 2

c) Geben Sie die Adjazenzmatrix A und die Wegematrix W an, die zu dem folgenden ungerichteten Graphen G mit 4 Knoten gehören:

G :


$$A = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \qquad W = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

/ 2

d) Es sei $A = \{a, b\}$, $L_1 = \{a^n \mid n \in \mathbb{N}_+\}$ und $L_2 = \{b^n \mid n \in \mathbb{N}_+\}$.

Geben Sie präzise die beiden folgenden Sprachen an:

- $L_1 \cdot L_2 =$ $\boxed{\{\mathtt{a}^n\mathtt{b}^m \mid n, m \in \mathbb{N}_+\}}$
- $(L_1^* \cdot L_2)^* =$ $\boxed{\text{alle Wörter aus } A^*\text{, die nicht mit } \mathtt{a} \text{ enden.}}$

Erläuterungen:

- *Falsch* ist $\{\mathtt{a}^n\mathtt{b}^n \mid n \in \mathbb{N}_+\}$, weil a-Block und b-Block nicht gleich lang sein müssen
- *Falsch* ist „alle Wörter, die mit b enden“, denn da fehlt das leere Wort. Aber es ginge z. B. `(a|b)*b|Ø*` oder Ähnliches.

/ 8 **Aufgabe 2** **(2 + 2 + 4 = 8 Punkte)**

/ 2

a) Es sei $A = \{\mathtt{a}, \mathtt{b}\}$ und die Abbildung $f\colon A^* \to A^*$ für $w \in A^*$ wie folgt gegeben:

$$f(w) = \mathtt{b}^{N_{\mathtt{a}}(w)+1}\mathtt{a}^{N_{\mathtt{b}}(w)+1}\,.$$

Dabei bezeichne $N_{\mathtt{a}}(w)$ bzw. $N_{\mathtt{b}}(w)$ die Anzahl der Vorkommen des Zeichen $\mathtt{a}$ bzw. $\mathtt{b}$ in w.

Vervollständigen Sie folgende induktive Definition für $g\colon A^* \to A^*$ so, dass für jedes $w \in A^*$ gilt: $f(w) = g(w)$.

$$g(\varepsilon) = \boxed{\mathtt{ba}}$$

$$\forall x \in A : \forall w \in A^* : g(xw) = \boxed{\begin{cases} \mathtt{b}g(w), & \text{falls } x = \mathtt{a} \\ g(w)\mathtt{a}, & \text{falls } x = \mathtt{b} \end{cases}}$$

Hinweis. Sie dürfen dabei keinen Bezug auf f nehmen.

/ 2

b) Beweisen Sie, dass die Abbildung f aus Teilaufgabe a) kein Homomorphismus ist.

$$\boxed{f(\varepsilon)f(\varepsilon) = \mathtt{baba} \neq \mathtt{ba} = f(\varepsilon) = f(\varepsilon \cdot \varepsilon)}$$

/ 4

c) Zeigen Sie durch vollständige Induktion, dass für alle $n \in \mathbb{N}_0$ gilt:

$$\sum_{k=0}^{n}(k+1)2^k = 1 + n2^{n+1}$$

Lösung 2

- Induktionsanfang (IA). Für $n = 0$ gilt:

$$\sum_{k=0}^{n}(k+1)2^k = 2^0 = 1 = 1 + n2^{n+1}$$

- Induktionsvoraussetzung (IV). Für ein beliebiges $n \in \mathbb{N}_0$ gelte $\sum_{k=0}^{n}(k+1)2^k = 1 + n2^{n+1}$.
- Induktionsschritt (IS). Dann gilt:

$$\begin{aligned}\sum_{k=0}^{n+1}(k+1)2^k &= (n+2)2^{n+1} + \sum_{k=0}^{n}(k+1)2^k \\ &\overset{\text{IV}}{=} (n+2)2^{n+1} + n2^{n+1} + 1 \\ &= (2n+2)2^{n+1} + 1 \\ &= (n+1)2^{n+2} + 1\end{aligned}$$

/ 6

Aufgabe 3 (1 + 1 + 1 + 1 + 2 = 6 Punkte)

Die Homomorphismen $h_1, h_2\colon \{0,1\}^* \to \{0,1\}^*$ seien wie folgt eindeutig festgelegt:

$$h_1(0) = 00 \qquad h_2(0) = 10$$
$$h_1(1) = 11 \qquad h_2(1) = 11$$

Ferner seien $f_1, f_2\colon \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ Abbildungen mit

$$f_1(x,y) = h_1(x)h_2(y), \qquad f_2(x,y) = h_2(x)00h_1(y)$$

für $x, y \in \{0,1\}^*$.

/ 1

a) Nennen Sie zwei Eigenschaften, die in der Vorlesung vorgestellt wurden, und die h_1 und Huffman-Codierungen gemeinsam haben.

Homomorphismus, Codierung (injektiv), ε-frei

/ 1

b) Geben Sie $f_2(01,01)$ an: 1011 00 0011

/ 1

c) Eine der Abbildungen f_1 und f_2 ist injektiv, die andere nicht. Welche ist *nicht* injektiv? Begründen Sie Ihre Antwort.

f_1 ist nicht injektiv. Zum Beispiel ist

$f_1(0,1) = h_1(0) \cdot h_2(1) = 0011 = 0011 \cdot \varepsilon = h_1(01) \cdot h_2(\varepsilon) = f_1(01,\varepsilon)$.

/ 1

d) Geben Sie eine unendliche Relation $M \subseteq (\{0,1\}^*)^4$ an so, dass für alle $(x_1, y_1, x_2, y_2) \in M$ gilt: $f_1(x_1, y_1) = f_2(x_2, y_2)$.

Zum Beispiel $M = \{(01^n, \varepsilon, \varepsilon, 1^n) \mid n \in \mathbb{N}_0\}$
denn $f_1(01^n, \varepsilon) = h_1(01^n) = 001^{2n} = 00h_2(1^n) = f_2(\varepsilon, 1^n)$
(Begründung war nicht verlangt)

/ 2

e) Es sei $M = \{w \mid w \in \{0,1\}^* \text{ und } |w| \geq 3\}$. Definieren Sie eine *injektive* Abbildung $f_3\colon M \times M \to \{0,1\}^*$ so, dass für alle $x \in M$ und $y \in M$ und für $i \in \{1,2\}$ gilt: $|f_3(x,y)| < |f_i(x,y)|$.

Z. B. $f_3(x,y) = h_2(x)00y$ oder $f_3(x,y) = x00h_2(y)$.

/ 7

Aufgabe 4 (1 + 2 + 2 + 2 = 7 Punkte)

In dieser Aufgabe geht es um gerichtete Graphen.

/ 1

a) Unter welchen Bedingungen ist $G = (V, E)$ ein gerichteter Graph?

V ist endliche nichtleere Menge und $E \subseteq V \times V$.

/ 2

b) Geben Sie *präzise* für jedes $n \in \mathbb{N}_+$ einen gerichteten Graphen $G_n = (V_n, E_n)$ an, der streng zusammenhängend ist und genauso viele Kanten wie Knoten hat.

$V_n = \boxed{\{i \in \mathbb{N}_0 \mid i < n\}}$

$E_n = \boxed{\{(i, i+1) \mid i, i+1 \in V_n\} \cup \{(n-1, 0)\}}$

/ 2

c) Begründen Sie, warum ein Graph mit mindestens zwei Knoten und weniger Kanten als Knoten nicht streng zusammenhängend sein kann.

/ 2

d) Für einen gerichteten Graphen $G = (V, E)$ sei $s_G \colon E \to V$ die Abbildung, die jeder Kante ihren Startknoten zuordnet. Beweisen oder widerlegen Sie:

> Für jeden gerichteten Graphen G gilt: Wenn s_G surjektiv ist, dann ist G streng zusammenhängend.

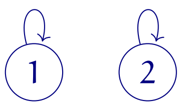
Lösung 4

c) Es sei $n \geq 2$ die Zahl der Knoten, $k < n$ die Zahl der Kanten und $z_G \colon E \to V$ mit $z_G((a, b)) = b$ die Abbildung, die jeder Kante Ihren Zielknoten zuordnet.

Da $k < n$ ist, ist z_G nicht surjektiv. Es gibt also einen Knoten y, zu dem keine Kante hinführt. Folglich führt überhaupt kein Weg in G von irgendwelchen Knoten $\neq y$ zu y hin. Und da $n \geq 2$ ist, gibt es einen Knoten $x \neq y$. Von x führt also kein Weg zu y.

Also ist G nicht zusammenhängend.

d) Die Aussage ist falsch; Gegenbeispiel:



| / 6 | **Aufgabe 5 (2 + 2 + 2 = 6 Punkte)**

Gegeben ist folgender Algorithmus A. Er bekommt als Eingabe ein Wort $x \in \{\mathtt{a}, \mathtt{b}\}^*$ und liefert ein Wort $w \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}^*$ als Ausgabe. $y, z \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}^*$ und $i \in \mathbb{N}_0$ werden von A als weitere Variablen benutzt. Es wird davon ausgegangen, dass die Symbole eines Wortes von links nach rechts ab 0 durchnummeriert sind; also ist z. B. $w(0)$ das erste Symbol eines nichtleeren Wortes w.

// x ist das Eingabewort
// $\cdot$ bedeutet immer Konkatenation
$y \leftarrow \varepsilon$
$z \leftarrow \mathtt{c}$
$i \leftarrow |x|$
while $i > 0$ **do**
  if $x(i-1) = \mathtt{a}$ **then** $y \leftarrow y \cdot \mathtt{bb}$ **fi**
  $z \leftarrow \mathtt{c} \cdot z$
  $i \leftarrow i - 1$
od
$w \leftarrow y \cdot z$
// w ist das Ausgabewort

| / 2 | a) Gegeben sei das Eingabewort $x = \mathtt{ab}$. Geben Sie folgende Werte an:

- Wert von z vor dem ersten Schleifendurchlauf: $z = \boxed{\mathtt{c}}$
- Anzahl r der Durchläufe durch den Schleifenrumpf: $r = \boxed{2}$
- Wert z_i von z am Ende des i-ten Schleifendurchlaufs für $1 \leq i \leq r$:

$z_1 = \boxed{\mathtt{cc}}$ $z_2 = \boxed{\mathtt{ccc}}$ $z_3 = \boxed{\text{———}}$ $z_4 = \boxed{\text{———}}$ $z_5 = \boxed{\text{———}}$

Hinweis: Nutzen Sie so viele Kästchen wie nötig und streichen Sie die restlichen durch.

- Wert von y am Ende des Algorithmus: $y = \boxed{\mathtt{bb}}$

| / 2 | b) Geben Sie eine kontextfreie Grammatik $G = (N, T, S, P)$ an, sodass $L(G)$ gleich der Menge aller möglichen Ausgaben w von A ist (für Eingaben $x \in \{\mathtt{a}, \mathtt{b}\}^*$). Ihre Grammatik darf höchstens 3 Nichtterminalsymbole haben (also $|N| \leq 3$).

| / 2 | c) Geben Sie eine präzise Beschreibung der Ausgabe w von A in Abhängigkeit der Eingabe $x \in \{\mathtt{a}, \mathtt{b}\}^*$ an.

Hinweis. Sie dürfen dabei die Notation $N_s(w)$ verwenden, welche für die Anzahl Vorkommen des Zeichens s im Wort w steht.

Lösung 5

b) $N = \{S, A\}$, $T = \{b, c\}$ und

$$P = \{\ S \rightarrow bbSc \mid Ac, \\ A \rightarrow Ac \mid \varepsilon\ \}$$

oder

$N = \{S\}$, $T = \{b, c\}$ und $P = \{S \rightarrow bbSc \mid Sc \mid c\}$

c) $z = (bb)^{N_a(x)} c^{|x|+1}$

/ 6

Aufgabe 6 (2 + 1 + 1 + 2 = 6 Punkte)

Es sei das Alphabet $X = \{\mathtt{a}, \mathtt{b}\}$ und die formale Sprache

$$L = \{w \in X^* \mid w \text{ enthält nicht } \mathtt{bab} \text{ als Teilwort}\}$$

gegeben.

/ 2

a) Geben Sie einen endlichen Akzeptor an, der L erkennt.

/ 1

b) Geben Sie einen regulären Ausdruck R an, sodass $L \cup \langle R \rangle = X^*$ gilt und $L \cap \langle R \rangle$ endlich aber nicht leer ist:

$R =$ `(a|b)*bab(a|b)*|∅*`

/ 1

c) Angenommen, R_1 ist ein regulärer Ausdruck mit $\langle R_1 \rangle = \overline{L}$, wobei $\overline{L}$ das Komplement von L bezeichne.

Gibt es einen regulären Ausdruck R_2, der R_1 als Teilwort enthält, und so, dass $\langle R_2 \rangle = L$ gilt? Falls ja, geben Sie einen solchen regulären Ausdruck an; andernfalls begründen Sie, warum das nicht sein kann.

Ja, z. B. $R_2 = \emptyset R_1$`|a*(b|baaa*)*a*`.

/ 2

d) Betrachten Sie jetzt für die oben angegebene Sprache L die Relation $\sim_L \subseteq X^* \times X^*$ mit

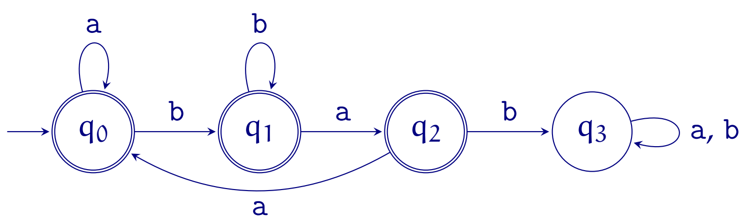
$$w_1 \sim_L w_2 \iff w_1 \in L \vee w_1 \cdot w_2 \notin L$$

für alle $w_1, w_2 \in X^*$.

Zeigen Sie, dass $\sim_L$ eine Äquivalenzrelation ist, und geben Sie die Anzahl der Äquivalenzklassen an.

Lösung 6

a)

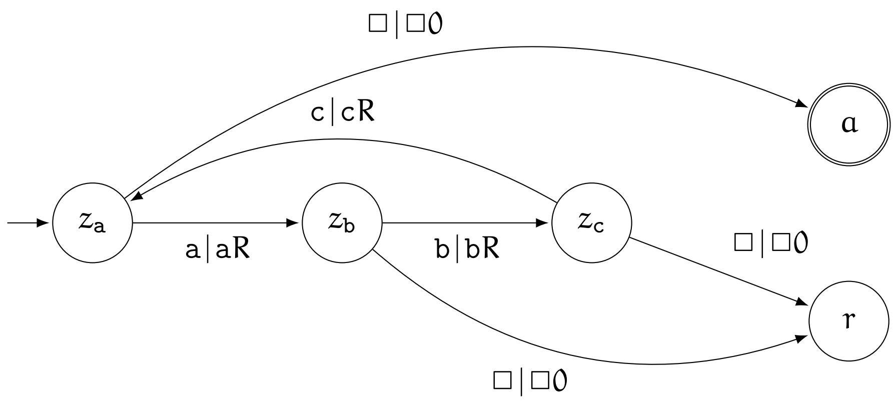


d) Die Bedingung $w_1 \in L \vee w_1 \cdot w_2 \notin L$ ist äquivalent zu folgender Aussage: „w_1 enthält bab als Teilwort $\rightarrow w_1 \cdot w_2$ enthält bab als Teilwort.“ Diese Aussage ist wahr für *alle* $w_1, w_2 \in X^*$, also ist $\sim_L = X^* \times X^*$ (trivialerweise) eine Äquivalenzrelation mit nur einer Äquivalenzklasse.

/ 5

Aufgabe 7 (1 + 1 + 3 = 5 Punkte)

Betrachten Sie folgende Turing-Maschine T mit Eingabealphabet $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$:



/ 1

a) Hält T hält für jede Eingabe an? Falls ja, begründen Sie, warum das so ist. Falls nicht, geben Sie eine Eingabe an, für die T nicht hält.

/ 1

b) Ist die folgende Aussage wahr oder falsch?

„Jede Turingmaschine, deren Zustandsgraph schlingenfrei und azyklisch ist, muss eine endliche Sprache akzeptieren.“

Beweisen Sie die Aussage oder widerlegen Sie sie anhand eines Gegenbeispiels.

/ 3

c) Die Zustandsübergänge von T seien wie folgt durchnummeriert:

1. $z_a \xrightarrow{\mathtt{a}|\mathtt{a}R} z_b$
2. $z_b \xrightarrow{\mathtt{b}|\mathtt{b}R} z_c$
3. $z_c \xrightarrow{\mathtt{c}|\mathtt{c}R} z_a$
4. $z_a \xrightarrow{\square|\square 0} a$
5. $z_b \xrightarrow{\square|\square 0} r$
6. $z_c \xrightarrow{\square|\square 0} r$

Geben Sie für jedes $i \in \{1, \dots, 6\}$ die Sprache $L(T_i)$ an, wobei T_i die Turingmaschine ist, die entsteht, wenn man den i-ten Zustandsübergang von T entfernt:

1. $L(T_1) = \{\varepsilon\}$
2. $L(T_2) = \{\varepsilon\}$
3. $L(T_3) = \{\varepsilon\}$
4. $L(T_4) = \{\}$
5. $L(T_5) = L(T)$
6. $L(T_6) = L(T)$

wobei $L(T) = \{(\mathtt{abc})^n \mid n \in \mathbb{N}_0\}$.

Lösung 7

a) Der Zustandsgraph von T enthält einen einzigen Zyklus, den T höchstens so viele Schritte entlangläuft, wie Eingabesymbole auf dem Band stehen. Da das nur endlich viele sind, kommt T nach endlich vielen Schritten aus dem Zyklus heraus, und hält höchstens einen Schritt später.

b) Die Aussage ist *falsch*. Z. B. hat folgende Turingmaschine keine Schlingen und keinen Zyklus, akzeptiert aber die Sprache X^*, wobei X das Eingabealphabet ist: