Lösungsvorschläge und Erläuterungen

Klausur zur Vorlesung
Grundbegriffe der Informatik
6. März 2017

Klausur-nummer				

Nachname:	
Vorname:	
Matr.-Nr.:	

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

Email-Adr.:		nur falls 2. Versuch

Aufgabe	1	2	3	4	5	6	
max. Punkte	1	8	10	11	8	12	
tats. Punkte							

Gesamtpunktzahl:			Note:	

/ 1

Aufgabe 1 (1 Punkt)

Beschriften Sie die Titelseite mit Ihrem Namen, Ihrer Matrikelnummer sowie Ihrer Klausurnummer. Geben Sie an, ob dies Ihr erster oder zweiter Versuch ist. Falls dieses Ihr zweiter Versuch ist, geben Sie bitte eine E-Mail Adresse an, unter der wir Sie erreichen können.

Beschriften Sie jedes weitere Blatt in der Kopfzeile der Vorderseite mit Ihrem Namen und Ihrer Matrikelnummer.

	/ 8

Aufgabe 2 **(1 + 1 + 1 + 2 + 1 + 1 + 1 = 8 Punkte)**

a) Gelten die Klammereinsparungsregeln der Aussagenlogik auch für die Prädikatenlogik? /1

Antwort: Ja

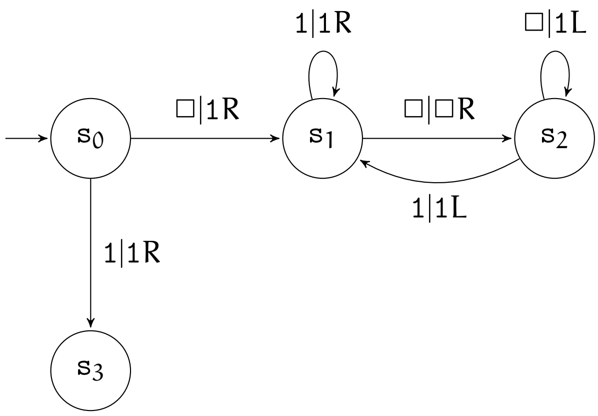
b) Listen Sie die in der Vorlesung eingeführten, aussagenlogischen Konnektive auf: /1

$\neg, \wedge, \vee, \rightarrow$

c) Hält die universelle Turingmaschine U immer? /1

Antwort: Nein

d) Ist diese graphisch angegebene Turingmaschine eine 3-Bibermaschine? /2



Antwort: Nein

e) Geben Sie die Anzahl der Knoten und Kanten des *gerichteten* Graphen an, der die folgende Adjazenzmatrix hat: /1

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Anzahl Knoten: 4

Anzahl Kanten: 8

f) Stimmt es, dass in einer prädikatenlogischen Formel eine gebundene Variable nicht frei vorkommen kann: /1

Antwort: Nein

g) Wieviele Wurzeln kann ein gerichteter Baum haben? /1

Antwort: 1

Weiterer Platz für Antworten zu Aufgabe 2:

/ 10 **Aufgabe 3 (4 + 6 = 10 Punkte)**

Die Fibonacci-Zahlen seien für alle $n \in \mathbb{N}_0$ wie folgt induktiv definiert:

$$\begin{aligned} F_0 &= 0 \\ F_1 &= 1 \\ \text{Für jedes } n \in \mathbb{N}_0 \text{ mit } n \geq 2: \quad F_n &= F_{n-1} + F_{n-2} \end{aligned}$$

a) Beweisen Sie die folgende Aussage durch vollständige Induktion: /4

$$\forall n \in \mathbb{N}_+ : \sum_{i=1}^{n} F_i = F_{n+2} - 1$$

b) Beweisen Sie die folgende Aussage durch vollständige Induktion: /6

$$\forall n \in \mathbb{N}_+ : \sum_{i=1}^{n} F_i^2 = F_n F_{n+1}$$

Lösung 3

a) **Induktionsanfang:** Es gilt für $n = 1$:

$$\sum_{i}^{n} F_i = \sum_{i}^{1} F_i = F_1 = 1 = 1 + 0 + 1 - 1 = F_2 + F_1 - 1 = F_3 - 1 = F_{n+2} - 1$$

Induktionsschritt: Es sei $n \in \mathbb{N}_+$ derart, dass

$$\sum_{i=1}^{n} F_i = F_{n+2} - 1 \qquad \text{(Induktionsvoraussetzung)}$$

Dann gilt

$$\sum_{i=1}^{n+1} F_i = \sum_{i=1}^{n} F_i + F_{n+1} \overset{\text{IV}}{=} F_{n+2} - 1 + F_{n+1} = F_{n+3} - 1$$

b) **Induktionsanfang:** Es gilt für $n = 1$:

$$\sum_{i}^{n} F_i^2 = \sum_{i}^{1} F_i^2 = F_1^2 = 1^2 = 1 \cdot (1 + 0) = F_1 F_2$$

Induktionsschritt: Es sei $n \in \mathbb{N}_+$ derart, dass

$$\sum_{i=1}^{n} F_i^2 = F_n F_{n+1} \qquad \text{(Induktionsvoraussetzung)}$$

Dann gilt

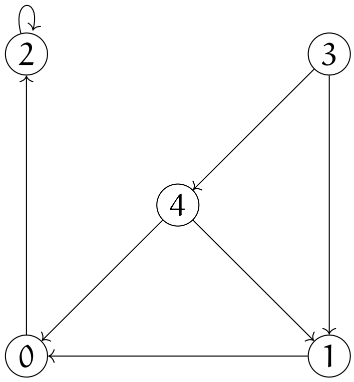
$$\begin{aligned}\sum_{i=1}^{n+1} F_i^2 &= \sum_{i=1}^{n} F_i^2 + F_{n+1}^2 \overset{\text{IV}}{=} F_n F_{n+1} + F_{n+1}^2 \\ &= F_{n+1}(F_{n+1} + F_n) \\ &= F_{n+1} F_{n+2}\end{aligned}$$

Platz für Antworten zu Aufgabe 3:

Weiterer Platz für Antworten zu Aufgabe 3:

/ 11	**Aufgabe 4** **(1 + 1 + 1 + 1 + 1 + 1 + 2 + 3 = 11 Punkte)**

Gegeben sei folgender gerichteter Graph G_1:



a) Geben Sie die Adjazenzmatrix von G_1 an. /1

b) Geben Sie die Wegematrix von G_1 an. /1

c) Ist G_1 azyklisch? Antwort: Nein /1

d) Ist G_1 streng zusammenhängend? Antwort: Nein /1

/1

e) Geben Sie den maximalen Eingangsgrad und den maximalen Ausgangsgrad der Knoten in G_1 an.

Maximaler Eingangsgrad: 2

Maximaler Ausgangsgrad: 2

/1

f) Geben Sie graphisch einen Teilgraph von G_1 an, der ein Baum ist und eine maximale Anzahl an Knoten und Kanten enthält.

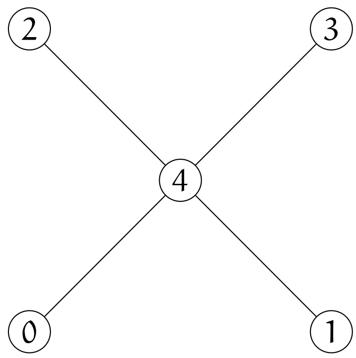
/2

g) Für jeden *ungerichteten* Graphen $G = (V, E)$ ist der sogenannte *Kantengraph* (engl. *line graph*) $L(G) = (V', E')$ wie folgt definiert: Wenn E nicht leer ist, dann ist

$$V' = E,$$
$$E' = \{\{e_1, e_2\} \mid e_1, e_2 \in E \text{ und } e_1 \cap e_2 \neq \emptyset\};$$

und wenn E leer ist, dann ist $V' = \{0\}$ und $E' = \{\}$.

Geben Sie für folgenden ungerichteten Graphen G_2 graphisch den Kantengraphen $L(G_2)$ an.



h) Ein ungerichter Graph G_S mit n Knoten habe die folgende $n \times n$ Adjazenzmatrix A: /3

$$A_{ij} = \begin{cases} 1 & \text{für } i = 0, j \neq 0 \\ 1 & \text{für } j = 0, i \neq 0 \\ 0 & \text{sonst} \end{cases}$$

Geben Sie die Adjazenzmatrix des zu G_S gehörenden Kantengraphen $L(G_S)$ an.

Lösung 4

a) $\begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix}$

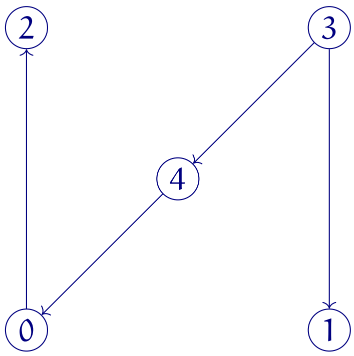
b) $\begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \end{pmatrix}$

c) s.o.

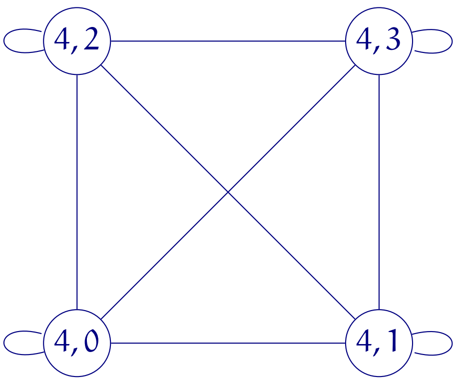
d) s.o.

e) s.o.

f)



g) Für jede Kante $e \in E$ gilt $e \cap e = e \neq \emptyset$. Also ist für jedes $e \in E$ auch $\{e, e\} = \{e\} \in E'$, weshalb im Kantengraphen alle Knoten eine Schlinge haben.



h) Für alle i und j ist $A_{ij} = 1$.

Weiterer Platz für Antworten zu Aufgabe 4:

/ 8

Aufgabe 5 (2 + 2 + 1 + 3 = 8 Punkte)

Es sei L_1 die formale Sprache

$$L_1 = L_a \cup L_b$$
$$L_a = \{w \mid w \in \{0,1,2,3,4,5\}^* \wedge \exists w_1, w_2 \in \{0,1\}^+ : w = w_1 4 w_2 5\}$$
$$L_b = \{w \mid w \in \{0,1,2,3,4,5\}^* \wedge \exists w_1, w_2 \in \{2,3\}^+ : w = w_1 4 w_2 5\}$$

/2

a) Geben Sie einen regulären Ausdruck R_1 derart an, dass $\langle R_1 \rangle = L_1$. Verwenden Sie in Ihrem regulären Ausdruck ausschließlich die Symbole $\mathtt{0}, \mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{4}, \mathtt{5}, (,), |, *, \emptyset$.

/2

b) Geben Sie graphisch einen endlichen Akzeptor an, der die formale Sprache L_1 akzeptiert.

Es sei L_2 die Sprache über dem Alphabet $\{\mathtt{a}, \mathtt{b}\}$, die genau diejenigen Wörter $w \in \{\mathtt{a}, \mathtt{b}\}^*$ enthält, für die gilt

- w beginnt mit zwei a,
- w endet mit ba,
- w enthält mindestens vier a und
- w enthält mindestens zwei b.

/1

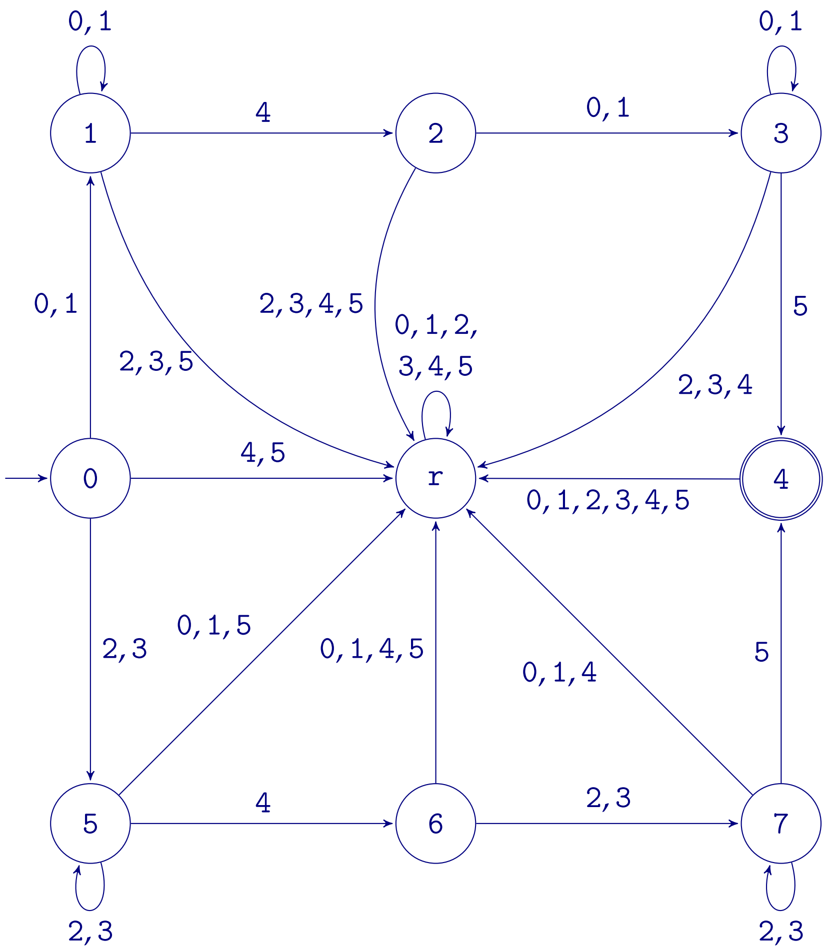
c) Geben Sie drei Wörter an, die zu L_2 gehören, und drei Wörter, die nicht zu L_2 gehören.

/3

d) Geben Sie einen regulären Ausdruck R_2 derart an, dass $\langle R_2 \rangle = L_2$. Verwenden Sie in Ihrem regulären Ausdruck ausschließlich die Symbole $\mathtt{a}, \mathtt{b}, (,), |, *, \emptyset$.

Lösung 5

a) R = `((0|1)(0|1)*4(0|1)(0|1)*5|((2|3)(2|3)*4(2|3)(2|3)*5))` b)



c) z.B. aaabba, aaaabba, aaaaabba sind in L_2 und ab, aab, aaa sind nicht in L_2.

d) $R =$ (aa)(a|b)∗(ab|ba)(a|b)∗(ba)

Platz für Antworten zu Aufgabe 5:

Weiterer Platz für Antworten zu Aufgabe 5:

/ 12

Aufgabe 6 **(2 + 2 + 2 + 2 + 2 + 2 = 12 Punkte)**

Eine *Double-ended queue* oder kurz *Deque* ist eine Datenstruktur, die eine Erweiterung des Stapels ist. Anders als beim Stapel können Daten an beiden Enden, dem vorderen und dem hinteren, der Deque gelesen und gespeichert werden. Die für eine Deque verfügbaren Operationen sind:

init Intitialisiert die Zeiger an den Speicherstellen 0 und 1

is_empty Gibt **w** zurück, falls die Deque leer ist, **f** sonst

push Einfügen eines Werts am hinteren Ende der Deque

pop Entfernen eines Werts am hinteren Ende der Deque, sofern die Deque nicht leer ist, sonst wird der Speicher unverändert gelassen

put Einfügen eines Werts am vorderen Ende der Deque

get Entfernen eines Werts am vorderen Ende der Deque, so fern die Deque nicht leer ist, sonst wird der Speicher unverändert gelassen

last liefert den Wert am hinteren Ende der Deque, ohne ihn aus der Deque zu entfernen, falls die Deque nicht leer ist, gibt ansonsten `undefiniert` zurück

first liefert den Wert am vorderen Ende der Deque, ohne ihn aus der Deque zu entfernen, falls die Deque nicht leer ist, gibt ansonsten `undefiniert` zurück

Wie in der Vorlesung vorgestellt, sei Adr eine Menge von Speicheradressen, Val eine Menge von Werten, die an den Speicheradressen abgelegt werden können, und somit $\mathtt{m}\colon \mathrm{Adr} \to \mathrm{Val}$ ein Speicherzustand. Somit ist $\mathrm{Mem} = \mathrm{Val}^{\mathrm{Adr}}$ die Menge aller möglichen Speicherzustände.

Für unsere Zwecke nehmen wir an, dass wir einen unendlich großen Speicher haben, der ausschließlich für eine Deque zur Verfügung steht und der positive und negative Speicheradressen kennt. D.h. $\mathrm{Adr} = \mathbb{Z}$. Außerdem nehmen wir an, dass in einer Speicherzelle jede beliebige Speicheradresse gespeicht werden kann, also $\mathrm{Val} \supseteq \mathbb{Z} \cup \{\mathtt{undefiniert}\}$.

Die Deque wird jetzt in diesem Speicher wie folgt repräsentiert: An Speicheradresse 0 steht die Speicheradresse, an der das nächste Element einer put Operation gespeichert werden kann. An Speicheradresse 1 steht die Speicheradresse, an der das nächste Element einer push Operation gespeichert werden kann. Bei put Operationen werden die Speicheradressen, an die das nächste Element derselben Operation gespeichert werden kann, immer kleiner; bei push Operationen werden die

Speicheradressen, an die das nächste Element derselben Operation gespeichert werden kann, immer größer.

Vervollständigen Sie analog zu den vorgegebenen Definitionen von init und last die restlichen Funktionsdefinitionen für die Realisierung der Operationen einer Deque mit Hilfe der aus der Vorlesung bekannten Operationen:

memread $\text{Mem} \times \text{Adr} \to \text{Val}$

$$(m, a) \mapsto m(a)$$

memwrite $\text{Mem} \times \text{Adr} \times \text{Val} \to \text{Mem}$

$$(m, a, v) \mapsto m'$$

$$\begin{aligned} \text{init}\colon \text{Mem} &\to \text{Mem}, \\ m &\mapsto \text{memwrite}(\text{memwrite}(m, 0, 2), 1, 3), \end{aligned}$$

$$\begin{aligned} \text{last}\colon \text{Mem} &\to \text{Val}, \\ m &\mapsto \begin{cases} \text{memread}(m, a') & \text{falls is\_empty}(m) = \mathbf{f} \\ \texttt{undefiniert} & \text{sonst} \end{cases} \\ &\quad \text{wobei } a' = \begin{cases} \text{memread}(m, 1) - 1 & \text{falls memread}(m, 1) \neq 2 \\ -1 & \text{sonst} \end{cases} \end{aligned}$$

$$\text{is\_empty}\colon \mathrm{Mem} \to \mathbb{B},$$

$$\mapsto$$

$$\text{push}\colon \mathrm{Mem} \times \mathrm{Val} \to \mathrm{Mem},$$

$$\mapsto$$

$$\text{pop}\colon \mathrm{Mem} \to \mathrm{Mem},$$

$$\mapsto$$

$$\text{put}: \text{Mem} \times \text{Val} \to \text{Mem},$$

$$\mapsto$$

$$\text{get}: \text{Mem} \to \text{Mem},$$

$$\mapsto$$

$$\text{first}: \text{Mem} \to \text{Val}$$

$$\mapsto$$

Lösung 6

$$\begin{aligned}
\mathrm{init}\colon \mathrm{Mem} &\to \mathrm{Mem},\\
m &\mapsto \mathrm{memwrite}(\mathrm{memwrite}(m,0,2),1,3),
\end{aligned}$$

$$\begin{aligned}
\mathrm{is_empty}\colon \mathrm{Mem} &\to \mathbb{B},\\
m &\mapsto \begin{cases} \mathbf{w}, & \text{falls } \mathrm{memread}(m,0) = -1 \text{ und } \mathrm{memread}(m,1) = 2,\\ \mathbf{w}, & \text{falls } \mathrm{memread}(m,0)+1 = \mathrm{memread}(m,1)\\ \mathbf{f}, & \text{sonst,} \end{cases}
\end{aligned}$$

$$\begin{aligned}
\mathrm{push}\colon \mathrm{Mem}\times\mathrm{Val} &\to \mathrm{Mem},\\
(m,v) &\mapsto \mathrm{memwrite}(m',1,a'),\\
&\quad \text{wobei } m' = \mathrm{memwrite}(m,\mathrm{memread}(m,1),v)\\
&\quad a' = \begin{cases} \mathrm{memread}(m,1)+1, & \text{falls } \mathrm{memread}(m,1) \neq -1\\ 2 & \text{sonst} \end{cases}
\end{aligned}$$

$$\begin{aligned}
\mathrm{pop}\colon \mathrm{Mem} &\to \mathrm{Mem},\\
m &\mapsto \begin{cases} \mathrm{memwrite}(m,1,a') & \text{falls } \mathrm{is_empty}(m) = \mathbf{f}\\ m & \text{sonst} \end{cases}\\
&\quad \text{wobei } a' = \begin{cases} \mathrm{memread}(m,1)-1 & \text{falls } \mathrm{memread}(m,1) \neq 2\\ -1 & \text{sonst} \end{cases}
\end{aligned}$$

$$\begin{aligned}
\mathrm{put}\colon \mathrm{Mem}\times\mathrm{Val} &\to \mathrm{Mem},\\
(m,v) &\mapsto \mathrm{memwrite}(m',0,a'),\\
&\quad \text{wobei } m' = \mathrm{memwrite}(m,\mathrm{memread}(m,0),v)\\
&\quad a' = \begin{cases} \mathrm{memread}(m,0)-1 & \text{falls } \mathrm{memread}(m,0) \neq 2\\ -1 & \text{sonst} \end{cases}
\end{aligned}$$

$$\mathrm{get}\colon \mathrm{Mem} \longrightarrow \mathrm{Mem},$$
$$m \mapsto \begin{cases} \mathrm{memwrite}(m, 0, a'), & \text{falls } \mathrm{is\_empty}(m) = \mathbf{f} \\ m & \text{sonst} \end{cases}$$
$$\text{wobei } a' = \begin{cases} \mathrm{memread}(m, 0) + 1 & \text{falls } \mathrm{memread}(m, 0) \neq -1 \\ 2 & \text{sonst} \end{cases}$$

$$\mathrm{last}\colon \mathrm{Mem} \longrightarrow \mathrm{Val},$$
$$m \mapsto \begin{cases} \mathrm{memread}(m, a') & \text{falls } \mathrm{is\_empty}(m) = \mathbf{f} \\ \texttt{undefiniert} & \text{sonst} \end{cases}$$
$$\text{wobei } a' = \begin{cases} \mathrm{memread}(m, 1) - 1 & \text{falls } \mathrm{memread}(m, 1) \neq 2 \\ -1 & \text{sonst} \end{cases}$$

$$\mathrm{first}\colon \mathrm{Mem} \longrightarrow \mathrm{Val}$$
$$m \mapsto \begin{cases} \mathrm{memread}(m, a') & \text{falls } \mathrm{is\_empty}(m) = \mathbf{f} \\ \texttt{undefiniert} & \text{sonst} \end{cases}$$
$$\text{wobei } a' = \begin{cases} \mathrm{memread}(m, 0) + 1, & \text{falls } \mathrm{memread}(m, 0) \neq -1 \\ 2 & \text{sonst} \end{cases}$$

Weiterer Platz für Antworten zu Aufgabe 6:

Weiterer Platz für Antworten zu Aufgabe 6: