Klausur zur Vorlesung
Grundbegriffe der Informatik
17. September 2018

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	7
max. Punkte	1	12	5	7	12	9	14
tats. Punkte							

Gesamtpunktzahl:			Note:	

/ 1

Aufgabe 1 (1 Punkt)

Beschriften Sie die Titelseite mit Ihrem Namen und Ihrer Matrikelnummer. 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.

/ 12 **Aufgabe 2** **(2 + 1 + 1 + 1 + 2 + 1 + 2 + 2 = 12 Punkte)**

a) Benennen Sie die Schlussregeln des Hilbert-Kalküls. /2

b) Gibt es Bibermaschinen, die nicht halten? /1

Antwort:

c) Geben Sie das Ergebnis des folgenden Ausdrucks an: $\{\} \times \{a, b\}$ /1

Antwort:

d) Was ist die Definition eines Datums gemäß Vorlesung? /1

e) Spezifizieren Sie mittels set comprehension die Menge M als die Menge aller natürlichen Zahlen, für die die folgenden Eigenschaften gelten: /2

- Sie sind größer oder gleich 0.
- Sie sind kleiner als 256.
- Sie sind eine Primzahl.

f) Ist die Wurzel eines ungerichteten Baumes durch seine Adjazenzmatrix eindeutig bestimmt? Antwort: /1

/2

g) Seien G und H Formeln der Aussagenlogik. Welche der folgenden Aussagenlogischen Formeln sind Tautologien und welche nicht?

a) $(G \to H) \leftrightarrow (\neg G \vee H)$

b) $G \wedge G \leftrightarrow G$

c) $G \to (G \wedge \neg G)$

d) $G \to (H \to G)$

Tautologien:

keine Tautologien:

/2

h) Geben Sie die Definition des Halteproblems an:

Weiterer Platz für Antworten zu Aufgabe 2:

/ 5

Aufgabe 3 **(3+2 = 5 Punkte)**

Wir haben in der Vorlesung ein Wort w über einem Alphabet A als surjektive Abbildung definiert. Gegeben sei das Alphabet $A = \{a, b, c, d\}$.

a) Geben Sie die Abbildung für das Wort $w_1 = abccba \in A^*$ formal an. /3

b) Geben Sie den Definitions- und Zielbereich der Abbildung für das leere Wort ϵ an. /2

Platz für Antworten zu Aufgabe 3:

/ 7

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

Fakultät ist eine Funktion, die einer natürlichen Zahl das Produkt aller natürlichen Zahlen, ohne Null, kleiner und gleich dieser Zahl zuordnet. Sie wird durch ein dem Argument nachgestelltes Ausrufezeichen abgekürzt, z.B.: 2!.
Hinweis: Das leere Produkt hat stets den Wert 1.

a) Geben Sie die Werte für 3! und 5! an. /1

b) Definieren Sie die Funktion Fakultät induktiv. /2

Gegeben sei folgende induktiv definierte Zahlenfolge:

$$a_0 = 5$$

Für jedes $n \in \mathbb{N}_0$: $a_{n+1} = a_n + 2n + 5$

/4

c) Beweisen Sie mittels vollständiger Induktion:
$a_n = (n+2)^2 + 1$ für alle $n \in \mathbb{N}_0$

Platz für Antworten zu Aufgabe 4:

/ 12	**Aufgabe 5 (2+2+1+1+3+3 = 12 Punkte)**

/2

a) Gegeben sei die folgende Adjazenzmatrix A

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

Zeichnen Sie einen Graphen für diese Adjazenzmatrix.

/2

b) Für eine Adjazenzmatrix B gelte:

$$B^2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Zeichnen Sie einen gerichteten Baum, dessen Adjazenzmatrix B die beschriebene Eigenschaft hat.

c) Geben Sie die Wegematrix C eines gerichteten Baumes mit 3 Knoten an, so dass $\sum_{i=0}^{2} \sum_{j=0}^{2} C_{ij}$ minimal ist.

/1

d) Geben Sie die Wegematrix D eines gerichteten Baumes mit 3 Knoten an, so dass $\sum_{i=0}^{2} \sum_{j=0}^{2} D_{ij}$ maximal ist.

/1

e) Sei W eine Wegematrix eines beliebigen, gerichteten Baumes mit n Knoten. Geben Sie ein $x \in \mathbb{N}$ und ein $y \in \mathbb{N}$ an, so dass folgende Abschätzung möglichst genau ist:
$x \leq \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} W_{ij} \leq y$

/3

/3

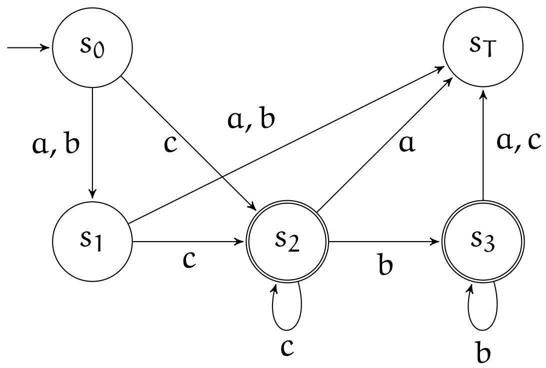
f) Gegeben sei die Wegematrix W eines beliebigen, gerichteten Baumes mit $n \in \mathbb{N}^+$ Knoten. Beweisen Sie folgende Aussage:
Es exisitert mindestens ein $k \in \mathbb{N}_0$ mit $k < n$ für das gilt:
$\sum_{i=0}^{n-1} W_{ik} = 1$

/ 9

Aufgabe 6 **(2+3+4 = 9 Punkte)**

/2

a) Gegeben sei der folgende endliche Akzeptor:



Geben Sie einen regulären Ausdruck an, der die Sprache beschreibt, welche der endliche Akzeptor akzeptiert.

b) Wandeln Sie den endlichen Akzeptor aus Aufgabe a) in einen Mealy-Automaten um. Die letzte Ausgabe des Mealy-Automaten soll eine 1 sein, falls der endliche Akzeptor aus a) das Wort akzeptiert. Ansonsten soll die letzte Ausgabe des Mealy-Automaten eine 0 sein. Definieren Sie den Mealy-Automaten vollständig für alle möglichen Wörter über dem Eingabealphabet $A = \{a, b, c\}$. Geben Sie den entsprechenden Mealy-Automaten grafisch an.

/3

c) Gegeben sei ein beliebiges, aber festes $k \in \mathbb{N}^+$. Geben Sie eine rechtslineare Grammatik G für die Sprache $L = \{a^n b^n | n \in N^+, n \leq k\}$ über dem Alphabet $A = \{a, b\}$ formal an.

/4

/ 14

Aufgabe 7 (1 + 1 + 1 + 1 + 1 + 1 + 2 + 2 + 2 + 2 = 14 Punkte)

Ein Lese-Cache ist eine besondere Form des Speichers in einer Rechnerarchitektur. Das Lesen aus einem solchen Cache ist deutlich schneller als aus dem Hauptspeicher. Dafür ist der Lese-Cache deutlich teurer als regulärer Speicher und ist daher deutlich kleiner gehalten als der Hauptspeicher.

Unser Lese-Cache, $cache\_mem$, soll jetzt ein Tupel aus zwei Speichern sein. Im ersten Speicher, $data\_mem$, werden die Werte aus dem Hauptspeicher $main\_mem$ abgelegt. Dabei bestimmt eine Hash-Funktion $hash$ an welcher Adresse j in $data\_mem$ der Wert an Adresse i in $main\_mem$ abgelegt wird. D.h., der Wert an Adresse i in $main\_mem$ wird an Adresse $j = hash(i)$ in $data\_mem$ abgelegt.

Der zweite Speicher, $addr\_mem$, speichert an Adresse i die Adresse j von $main\_mem$, deren zugehöriger Wert in $main\_mem$ zur Zeit an Adresse i in $data\_mem$ abgelegt ist. Der Wert von $addr\_mem$ an Adresse i ist gleich -1, sollte an Adresse i in $data\_mem$ noch kein Wert aus dem Hauptspeicher stehen.

Die verfügbaren Operationen des Lese-Cache sind:

main_memread $addr$: Lese den Wert an Adresse $addr$ im Hauptspeicher $main\_mem$.

main_memwrite $addr$ val : Schreibe den Wert val an die Adresse $addr$ im Hauptspeicher $main\_mem$.

data_memread $addr$: Lese den Wert an Adresse $addr$ im Speicher $data\_mem$.

data_memwrite $addr$ val : Schreibe den Wert val an die Adresse $addr$ im Speicher $data\_mem$.

addr_memread $addr$: Lese den Wert an Adresse $addr$ im Speicher $addr\_mem$.

addr_memwrite $addr$ val : Schreibe den Wert val an die Adresse $addr$ im Speicher $addr\_mem$.

reset $addr$: Setze den Lese-Cache so, dass an Adresse $addr$ in $data\_mem$ kein Wert aus dem Hauptspeicher steht

is_in_cache $addr$: Gebe **w** zurück, falls der Wert an Adresse $addr$ des Hauptspeichers im Cache ist, **f** sonst cacheread $addr$: Lese den Wert, der an Adresse $addr$ im Hauptspeicher steht, aus dem Cache, falls der Wert im Cache ist. Sonst lese den Wert aus dem Hauptspeicher und lege ihn im Lese-Cache ab.

cachewrite $addr\ val$: Schreibe den Wert val an die Adresse $addr$ im Hauptspeicher $main\_men$ und erhalte gleichzeitig die Integrität des Cache.

Ähnlich wie in der Vorlesung vorgestellt, sei Adr_$Main$ die Menge der Speicheradressen des Hauptspeichers, Adr_$Cache$ die Menge der Speicheradressen von $data\_mem$ und von $addr\_mem$. Ferner sei Val die Menge von Werten, die im Hauptspeicher abgelegt werden können.

Die drei Speicher $data\_mem$, $addr\_mem$ und $main\_mem$ seien jetzt mathematisch als Abbildungen modelliert, so wie in der Vorlesung für Speicher eingeführt, mit Definitions- und Zielmengen, die zur beschriebenen Funktion und Aufgabe der drei Speicher passen. D.h.:

- $data\_mem \in \mathrm{Val}^{\mathrm{Adr\_}Cache}$
- $addr\_mem \in \mathrm{Adr\_}Main^{\mathrm{Adr\_}Cache}$
- $main\_mem \in \mathrm{Val}^{\mathrm{Adr\_}Main}$

Ihre Aufgabe ist es jetzt, folgenden Funktionsdefinitionen für die Realisierung der Operationen des Lese-Cache zu vervollständigen. Gehen Sie dazu analog vor wie bei den aus der Vorlesung bekannten Operationen für allgemeinen Speicher. Denken Sie daran, sowohl Definitions- und Zielbereich der Funktionen zu spezifizieren als auch die Abbildungsvorschrift.

main_memread: $\rightarrow$

$\mapsto$

main_memwrite: $\rightarrow$

$\mapsto$

data_memread: $\rightarrow$

$\mapsto$

data_memwrite: $\rightarrow$

$\mapsto$

$$
\begin{aligned}
\text{addr\_memread}: &\quad \rightarrow \\
&\quad \mapsto \\
\text{addr\_memwrite}: &\quad \rightarrow \\
&\quad \mapsto \\
\text{reset}: &\quad \rightarrow \\
&\quad \mapsto \\
\text{is\_in\_cache}: &\quad \rightarrow \\
&\quad \mapsto
\end{aligned}
$$

$$\text{cacheread}: \quad \longrightarrow$$

$$\longmapsto$$

$$\text{cachewrite}: \quad \longrightarrow$$

$$\longmapsto$$

Weiterer Platz für Antworten zu Aufgabe 7: