Karlsruher Institut für Technologie

Institut für Angewandte Informatik
und Formale Beschreibungsverfahren

20.07.2015

Lösung zur Klausur über den Stoff der Vorlesung „Grundlagen der Informatik II“ (90 Minuten)

Name: _____ Vorname: _____

Matr.-Nr.: _____ Semester: _____ (SS 2015)

Ich bestätige, dass ich die folgenden Angaben gelesen und mich von der Vollständigkeit dieses Klausurexemplars überzeugt habe (Seiten 1-25).

_____

Unterschrift des o. g. Klausurteilnehmers bzw. der o. g. Klausurteilnehmerin

Anmerkungen:

1. Legen Sie bitte Ihren Studierendenausweis bereit.
2. Bitte tragen Sie **Name**, **Vorname** und **Matr.-Nr.** deutlich lesbar ein.
3. Die folgenden **11 Aufgaben** sind vollständig zu bearbeiten.
4. Folgende Hilfsmittel sind zugelassen: **keine.**
5. Täuschungsversuche führen zum Ausschluss von der Klausur.
6. Unleserliche oder mit Bleistift geschriebene Lösungen können von der Klausur bzw. Wertung ausgeschlossen werden.
7. Die Bearbeitungszeit beträgt 90 Minuten.

Nur für den Prüfer :

1	2	3	4	5	6	7	8	9	10	11	-	-	-	-	-	gesamt
(9)	(10)	(8)	(8)	(8)	(9)	(8)	(10)	(8)	(8)	(4)						(90)

Aufgabenübersicht

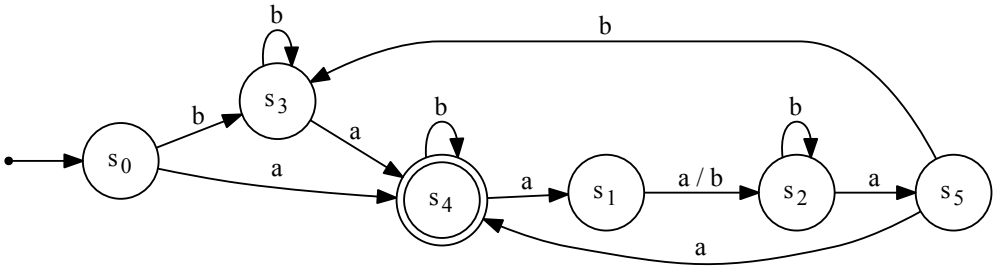
Aufgabe 1 **9 Punkte**

2015-N-01 **Endliche Automaten**

/ 9

Gegeben sei folgender deterministischer endlicher Automaten $A = (E, S, \delta, s_0, F)$. Durch das abgebildete Zustandsüberführungsdiagramm sei δ definiert.

δ :



(a) Minimieren Sie A und geben Sie den minimierten Automaten A' vollständig an. Geben Sie insbesondere die Minimierungstabelle und ein Zustandsüberführungsdiagramm δ' an.

/ 5

Lösung:

- Übergangstabelle für A:

	a	b
s_0	s_4	s_3
s_1	s_2	s_2
s_2	s_5	s_2
s_3	s_4	s_3
s_4	s_1	s_4
s_5	s_4	s_3

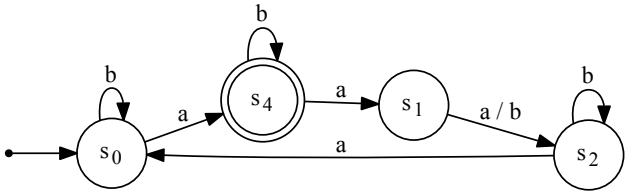
- Dreieckstabelle:

s_1	$\times_1$				
s_2	$\times_1$	$\times_2$			
s_3	$-$	$\times_1$	$\times_1$		
s_4	$\times_0$	$\times_0$	$\times_0$	$\times_0$	
s_5	$-$	$\times_1$	$\times_1$	$-$	$\times_0$
	s_0	s_1	s_2	s_3	s_4

- Übergangstabelle für A':

	a	b
s_0	s_4	s_0
s_1	s_2	s_2
s_2	s_0	s_2
s_4	s_1	s_4

- Der minimierte Automat A' lautet wie folgt: $A' = (\{a, b\}, \{s_0, s_1, s_2, s_4\}, \delta', s_0, \{s_4\})$



(b) Geben Sie die Mengen aller zueinander k-äquivalenten Zustände für $k \in \{0, 1, 2\}$ und die Mengen äquivalenter Zustände des endlichen Automaten A an. Verwenden Sie hierfür die nachfolgende Tabelle. Geben Sie auch einelementige Mengen an.

/ 4

Lösung:

0−Äquivalenz	$\{s_0, s_1, s_2, s_3, s_5\}, \{s_4\}$
1−Äquivalenz	$\{s_0, s_3, s_5\}, \{s_1, s_2\}, \{s_4\}$
2−Äquivalenz	$\{s_0, s_3, s_5\}, \{s_1\}, \{s_2\}, \{s_4\}$
Äquivalenz	$\{s_0, s_3, s_5\}, \{s_1\}, \{s_2\}, \{s_4\}$

Skript für den Ausgangsautomaten (http://www.xwizard.de)

```
fsm:
(s0, a) | (s3, a) | (s4, b) | (s5, a) => s4;
(s0, b) | (s3, b) | (s5, b) => s3;
(s1, a) | (s1, b) | (s2, b) => s2;
(s2, a) => s5;
(s4, a) => s1;
--declarations--
s0=s0;
F=s4
--declarations-end--
```

Skript für den minimierten Automaten (http://www.xwizard.de)

```
fsm:
(s0, a) | (s4, b) => s4;
(s0, b) | (s2, a) => s0;
(s1, a) | (s1, b) | (s2, b) => s2;
(s4, a) => s1;
--declarations--
s0=s0;
F=s4
--declarations-end--
```

Aufgabe 2 **10 Punkte**

2015-N-02 **Grammatiken**

/ 10

Gegeben sei die folgende Grammatik – beachten Sie, dass „(“ und „)“ Terminalzeichen sind:

$$G = (\underbrace{\{A_1, A_2, A_3, A_5, A_{10}\}}_{N}, \underbrace{\{(,), \star, +, \emptyset, a, b, c, \cdot\}}_{T}, P, \underbrace{A_5}_{S})$$

mit

$$\begin{aligned} P = \{&A_5 \to a \mid b \mid c \mid \emptyset \mid (A_5 A_1 A_5 A_3 \mid (A_5 A_{10} A_5 A_3 \mid (A_5 A_3 A_2, \\ &A_1 \to +, \\ &A_2 \to \star, \\ &A_3 \to), \\ &A_{10} \to \cdot\} \end{aligned}$$

Spielen Sie auch mit diesem Skript für den XWizard herum.

```
grammar parse(XkaX,XkaX,a,XplusX,b,XkzX,XkzX,*)--0:
A(1) => XplusX;
A(2) => *;
A(3) => XkzX;
A(5) => 0 | a | b | c | XkaX, A(5), A(1), A(5), A(3)
| XkaX, A(5), A(10), A(5), A(3) | XkaX, A(5), A(3), A(2);
A(10) => x;
A(5) => a, B | b, B | c, B;
B => XplusX, A(5), B | XplusX, A(5);
--declarations--
N=A(1),A(2),A(3),A(5),A(10),B;
S=A(5);
multiLetterSymbolsHaveIndex=true;
maxdepth=5;
cutNonTerminalBranches=true;
cutTerminalDoubleBranches=true;
maxLengthWords=6
--declarations-end--
```

(a) Von welchem Chomsky-Typ / welchen Chomsky-Typen ist die Grammatik?

/ 1

Lösung: Von den Typen 0, 1, 2.

(b) In welcher Normalform ist die Grammatik?

/ 1

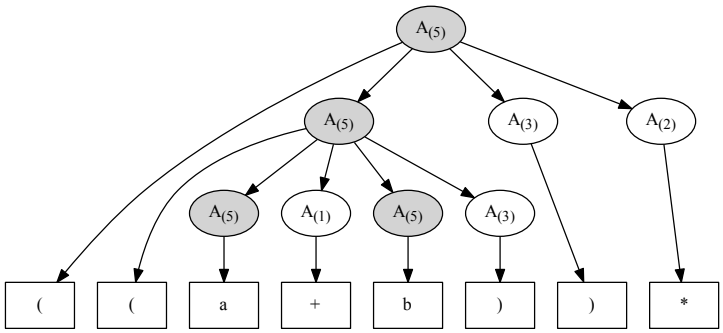
Lösung: In der Greibach-Normalform.

(c) Leiten Sie das Wort $((a + b))^\star$ bzgl. G ab (Ableitungsfolge oder Ableitungsbaum).

/ 3

Lösung:

$$\begin{aligned}\overline{A_5} &\Rightarrow \underline{(\overline{A_5}A_3A_2} \Rightarrow \underline{((\overline{A_5}A_1A_5A_3}A_3A_2 \Rightarrow ((\underline{a}\overline{A_1}A_5A_3A_3A_2\\ &\Rightarrow ((a\underline{+}\overline{A_5}A_3A_3A_2 \Rightarrow ((a + \underline{b}\overline{A_3}A_3A_2 \Rightarrow ((a + b)\underline{\overline{A_3}}A_2 \Rightarrow ((a + b)\underline{\overline{A_2}}\\ &\Rightarrow ((a + b))^\star\end{aligned}$$



G erzeugt die Sprache aller regulären Ausdrücke (RA) über der Basismenge $E = \{a, b, c\}$ **ohne abkürzende Schreibweisen** (d. h. die RA sind vollständig korrekt geklammert).

(d) Geben Sie eine zusätzliche Regel r für G an, sodass auch abkürzende Schreibweisen für den Plus-Operator möglich werden, also $x + y$ für $(x + y)$ wobei x, y RA sind.

Hinweis: r muss (zunächst) nicht in Normalform sein, wie die übrigen Regeln.

/ 2

Lösung: $P' = P \cup \{A_5 \to A_5 + A_5\}$.

(e) Wandeln Sie die Regel r aus (d) in die Normalform der übrigen Regeln um.

Hinweis: Sie werden mehr als eine neue Regel benötigen.

/ 3

Lösung:

$$P'' = P \cup \{A_5 \to aB \mid bB \mid cB,$$
$$B \to +A_5B \mid +A_5\}$$

Aufgabe 3 **8 Punkte**

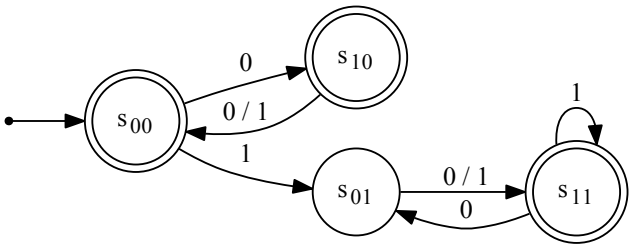
2015-N-03 **Reguläre Ausdrücke**

/ 8

Gegeben sei der endliche Automat

$$EA = (\{0, 1\}, \{s_{00}, \ldots, s_{11}\}, \delta, s_{00}, \{s_{00}, s_{10}, s_{11}\})$$

δ:



Spielen Sie auch mit diesem Skript für den XWizard herum.

```
fsm:
(s00, 0) => s10;
(s00, 1) | (s11, 0) => s01;
(s01, 0) | (s01, 1) | (s11, 1) => s11;
(s10, 0) | (s10, 1) => s00;
--declarations--
s0=s00;
F=s00,s10,s11
--declarations-end--
```

(a) Geben Sie einen regulären Ausdruck (RA) α an mit $L(\alpha) = L(EA)$.

/ 5

Lösung: $\alpha = (0(0 + 1))^\star + 0((0 + 1)0)^\star + (0(0 + 1))^\star 1(0 + 1)(1 + 0(0 + 1))^\star$

(b) Angenommen, Sie sollen für zwei gegebene RA α_1, α_2 herausfinden, ob sie äquivalent sind, ob also $L(\alpha_1) = L(\alpha_2)$. Wie würden Sie das mit den in der Vorlesung vorgestellten Mitteln bewerkstelligen? Erläutern Sie Ihre Vorgehensweise.

Hinweis: Überlegen Sie, welche zu RA äquivalenten Konstrukte wir kennen, deren Äquivalenz getestet werden kann.

/ 3

Lösung: Ein Verfahren, das direkt auf RA arbeitet, haben wir nicht kennengelernt. Wir könnten aber beide Ausdrücke zunächst in endliche Automaten umwandeln – diese wären evtl. nichtdeterministisch. Mit dem Potenzmengen-Verfahren könnten wir sie deterministisch machen und dann minimieren. Da minimale deterministische Automaten (bis auf Isomorphien) eindeutig sind, könnten wir die entstehenden Automaten vergleichen und genau dann „JA“ ausgeben, wenn sie identisch sind.

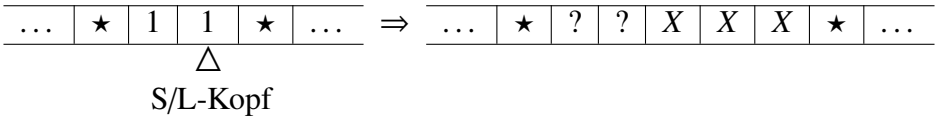
Aufgabe 4 **8 Punkte**

2015-N-04 **Turingmaschinen**

/ 8

Geben Sie eine Turingmaschine $T = (E, B, S, \delta, s_0, F)$ an, die k-mal das Symbol X rechts neben die Bandinschrift schreibt, wobei k dem Wert der als Dualzahl interpretierten Bandinschrift entspricht.

Beispiel (Fragezeichen stehen für beliebige Zeichen aus dem Bandalphabet):



Hinweise:

- Zur Vereinfachung steht der Schreib-/Lesekopf (S/L-Kopf) zunächst über dem Zeichen, das am weitesten **rechts und nicht links** (wie sonst in der Vorlesung üblich) auf der Bandinschrift steht (siehe Beispiel).
- Die Bandinschrift darf von Ihrer Turingmaschine beliebig verändert werden.
- Sie können die Dualzahl auf der Bandinschrift in jeder Iteration um 1 verringern, um eine Schleife zum Schreiben der X zu implementieren.
- Die Zustandsübergangstabelle muss nicht vollständig ausgenutzt werden.

Lösung:

$T = (\{0, 1\}, \{0, 1, X, \star\}, \{s_0, s_1, s_e\}, \delta, s_0, \{s_e\})$

	0	1	X	$\star$	
s_0	$(s_0, 0, L)$	$(s_1, 0, R)$	(s_0, X, L)	$(s_e, \star, N)$	Linkslaufen, dekrementieren
s_1	$(s_1, 1, R)$		(s_1, X, R)	(s_0, X, L)	Übertrag, rechtslaufen, X schreiben
s_e					Endzustand
s_x	$(s_x, 0, R)$	$(s_x, 1, R)$		$(s_0, \star, L)$	Alt. Startzustand zum Rechtslaufen

Spielen Sie auch mit diesem Skript für den XWizard herum.

```
turing:
(s0, *) => (se, *, N);
(s0, 0) => (s0, 0, L);
(s0, 1) => (s1, 0, R);
(s0, X) | (s1, *) => (s0, X, L);
(s1, 0) => (s1, 1, R);
```

```
(s1, X) => (s1, X, R);
(sx, *) => (s0, *, L);
(sx, 0) => (sx, 0, R);
(sx, 1) => (sx, 1, R);
--declarations--
s0=sx;
F=se;
blank=*;
inputs=11;
runStepsScript=100;
shortTrace=false
--declarations-end--
```

Bandinhalt	Zustand
$\hat{1}1$	$(s_x, 1) \to (s_x, 1, R)$
$1\hat{1}$	$(s_x, 1) \to (s_x, 1, R)$
$11\hat{\star}$	$(s_x, \star) \to (s_0, \star, L)$
$1\hat{1}\star$	$(s_0, 1) \to (s_1, 0, R)$
$10\hat{\star}$	$(s_1, \star) \to (s_0, X, L)$
$1\hat{0}X$	$(s_0, 0) \to (s_0, 0, L)$
$\hat{1}0X$	$(s_0, 1) \to (s_1, 0, R)$
$0\hat{0}X$	$(s_1, 0) \to (s_1, 1, R)$
$01\hat{X}$	$(s_1, X) \to (s_1, X, R)$
$01X\hat{\star}$	$(s_1, \star) \to (s_0, X, L)$
$01\hat{X}X$	$(s_0, X) \to (s_0, X, L)$
$0\hat{1}XX$	$(s_0, 1) \to (s_1, 0, R)$
$00\hat{X}X$	$(s_1, X) \to (s_1, X, R)$
$00X\hat{X}$	$(s_1, X) \to (s_1, X, R)$
$00XX\hat{\star}$	$(s_1, \star) \to (s_0, X, L)$
$00X\hat{X}X$	$(s_0, X) \to (s_0, X, L)$
$00\hat{X}XX$	$(s_0, X) \to (s_0, X, L)$
$0\hat{0}XXX$	$(s_0, 0) \to (s_0, 0, L)$
$\hat{0}0XXX$	$(s_0, 0) \to (s_0, 0, L)$
$\hat{\star}00XXX$	$(s_0, \star) \to (s_e, \star, N)$
$\hat{\star}00XXX$	

Aufgabe 5	**8 Punkte**
2015-N-05	**Komplexität**

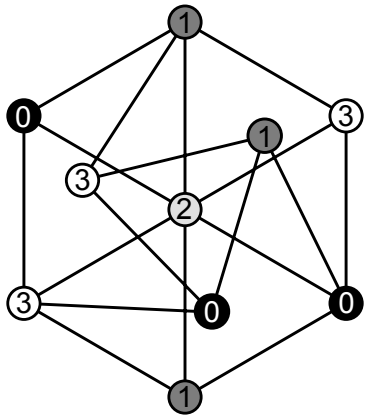
/ 8

k-COLOR bezeichnet das Entscheidungsproblem, ob die Knoten eines Graphen mit $k \geq 3$ Farben so eingefärbt werden können, dass benachbarte Knoten jeweils unterschiedliche Farben besitzen.

Formal wird dies wie folgt ausgedrückt: Sei $G = (V, E)$ ein ungerichteter Graph mit Knotenmenge V und Kantenmenge E. Die Funktion $f : V \to C \subseteq \mathbb{N}_0$ mit $|C| = k$ heißt Knotenfärbung von G. (C beschreibt hierbei eine Menge an Farben, die durch natürliche Zahlen kodiert sind.) f heißt **zulässig**, falls

$$\forall (v, w) \in E : f(v) \neq f(w).$$

Beispiel einer Graphfärbung für $k = 4$:



(a) Nehmen Sie an, Sie wüssten, dass k-COLOR NP-vollständig ist, und wollten zeigen, dass SAT (Entscheidungsproblem der Aussagenlogik) NP-schwer ist. Beschreiben Sie konkret, welche Schritte erforderlich sind, um k-COLOR auf SAT zu reduzieren.

/ 2

Lösung:

Es muss gezeigt werden, dass jede k-COLOR Probleminstanz als aussagenlogische Formel in konjunktiver Normalform dargestellt werden kann und die Transformation in polynomieller Zeit möglich ist. Für einen gegebenen Graphen existiert genau dann eine zulässige k-Färbung, wenn diese aussagenlogische Formel erfüllbar ist.

Nehmen Sie im Folgenden an, dass es für jeden Knoten v und jede Farbe $c \in C$ eine Boolesche Variable v_c gibt, die genau dann wahr ist, wenn v die Farbe c annimmt.

(b) Wie können Sie durch eine aussagenlogische Formel in konjunktiver Normalform darstellen, dass jeder Knoten mindestens einen Farbwert annehmen muss? Geben Sie diese Normalform formal oder umgangssprachlich präzise an.

Hinweis: Ignorieren Sie hierbei, ob die resultierende Färbung des Graphen zulässig ist.

/ 2

Lösung:

Sei $C = \{1, \ldots, k\}$. Für jeden Knoten v wird eine Klausel $K_v := (v_1 \vee \ldots \vee v_k)$ erzeugt. Die einzelnen Klauseln K_v werden für alle $v \in V$ durch logische UND-Operatoren miteinander verknüpft.

(c) Wie können Sie durch eine aussagenlogische Formel in konjunktiver Normalform die Zulässigkeit von f sicherstellen, dass also keine benachbarten Knoten mit gleicher Färbung existieren? Geben Sie diese Normalform formal oder umgangssprachlich präzise an.

Hinweis: Ignorieren Sie hierbei zunächst, ob jeder Knoten überhaupt eine Farbe annimmt und fügen Sie dann das Ergebnis aus (b) hinzu.

/ 4

Lösung:

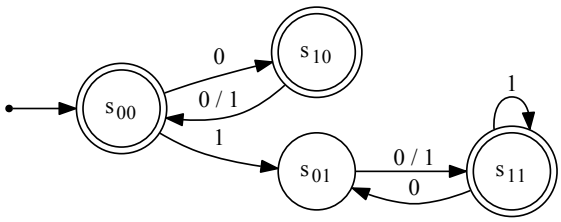
Für alle benachbarten Knotenpaare (v, w) muss gelten $\neg(v_i \wedge w_i)$ für alle $i \in C$. Jene Formulierung ist äquivalent zu $(\neg v_i \vee \neg w_i)$. Diese Klauseln werden für alle $i \in C$ und alle Knotenpaare (v, w) durch logische AND-Operatoren miteinander verknüpft, um eine aussagenlogische Formel in konjunktiver Normalform zu erhalten. Diese Formel mit der Formel aus Aufgabenteil (b) per AND-Operator verknüpft.

Aufgabe 6 **9 Punkte**

2015-N-06 **Schaltwerke**

/ 9

Aus Aufgabe 3 sei wieder $EA = (\{0,1\}, \{s_{00}, \ldots, s_{11}\}, \delta, s_{00}, \{s_{00}, s_{10}, s_{11}\})$ gegeben mit δ:



Das u. a. Schaltwerk soll für fortlaufende Eingaben (an E) das Verhalten von EA simulieren.

(a) Vervollständigen Sie das Schaltwerk so, dass die beiden Flipflops ($RS1, RS0$) genau dann die Werte $(x, y) \in \mathbb{B}^2$ enthalten, wenn sich der endliche Automat bei der entsprechenden Eingabesequenz in Zustand s_{xy} befindet.

Hinweis: Die gestrichelten bzw. gepunkteten Leitungen repräsentieren die Übergänge $s_{10} \overset{0/1}{\rightarrow} s_{00}$ und $s_{01} \overset{0/1}{\rightarrow} s_{11}$. Überlegen Sie für sich, warum E dabei nicht verbunden ist und warum das gepunktete AND-Gatter, welches zu s_{10} gehört, unnötig ist. Ergänzen Sie für Ihre Lösung nur Ein- bzw. Ausgänge der grauen Gatter und der beiden Flipflops. Nutzen Sie zusätzliche NOT-Gatter.

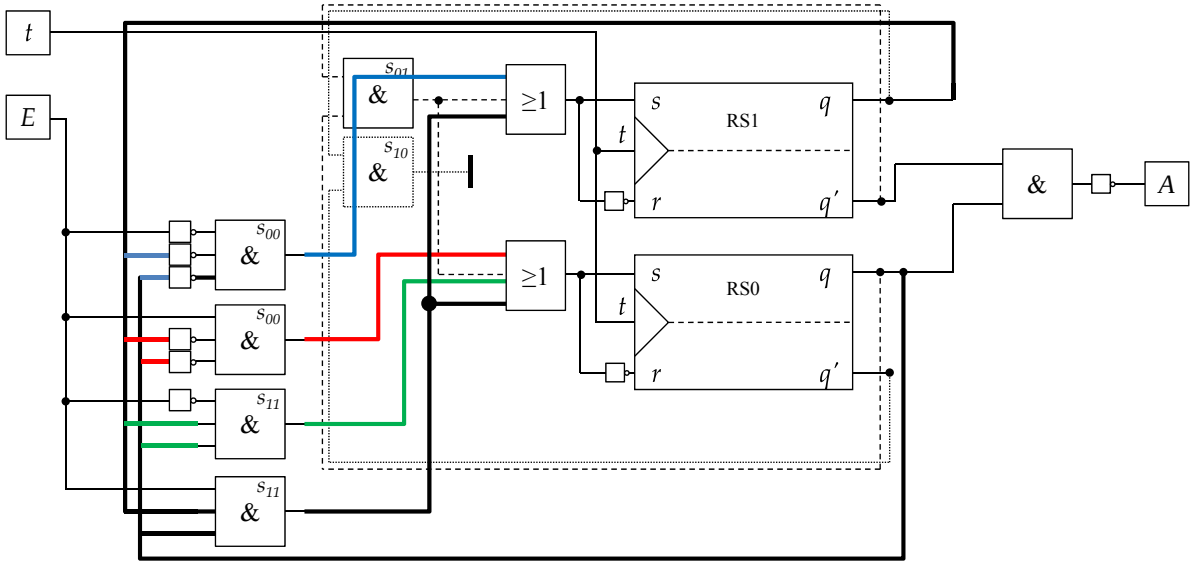
/ 7

So kommen wir auf **6,5 Punkte**. Dazu (jew.!) **0,5 Punkte**, wenn die Verbindungen von den Flipflops zu den ANDs bzw. von den ANDs zu den ORs **vollständig richtig** sind. (Natürlich insgesamt maximal **7 Punkte**.)

(b) Verbinden Sie die Flipflops mit dem Ausgang A, sodass genau dann 1 ausgegeben wird, wenn s_{xy} ein Endzustand ist. Nutzen Sie zusätzliche Gatter (AND, OR und/oder NOT).

/ 2

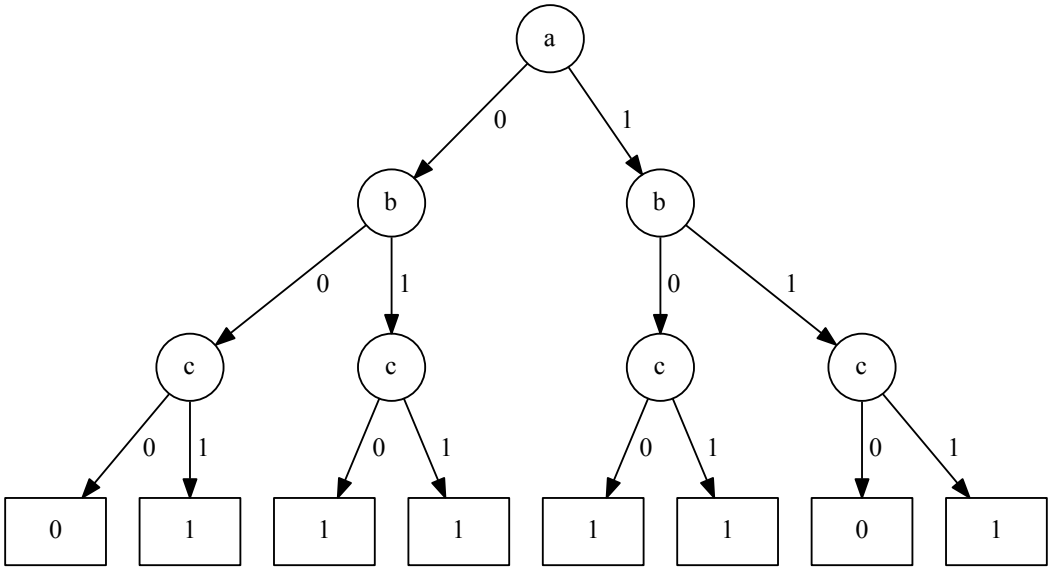
Lösung:



Aufgabe 7	**8 Punkte**
2015-N-7	**BDD**

/ 8

Gegeben sei die durch den abgebildeten Baum definierte Boolesche Funktion $f : \mathbb{B}^3 \to \mathbb{B}$.

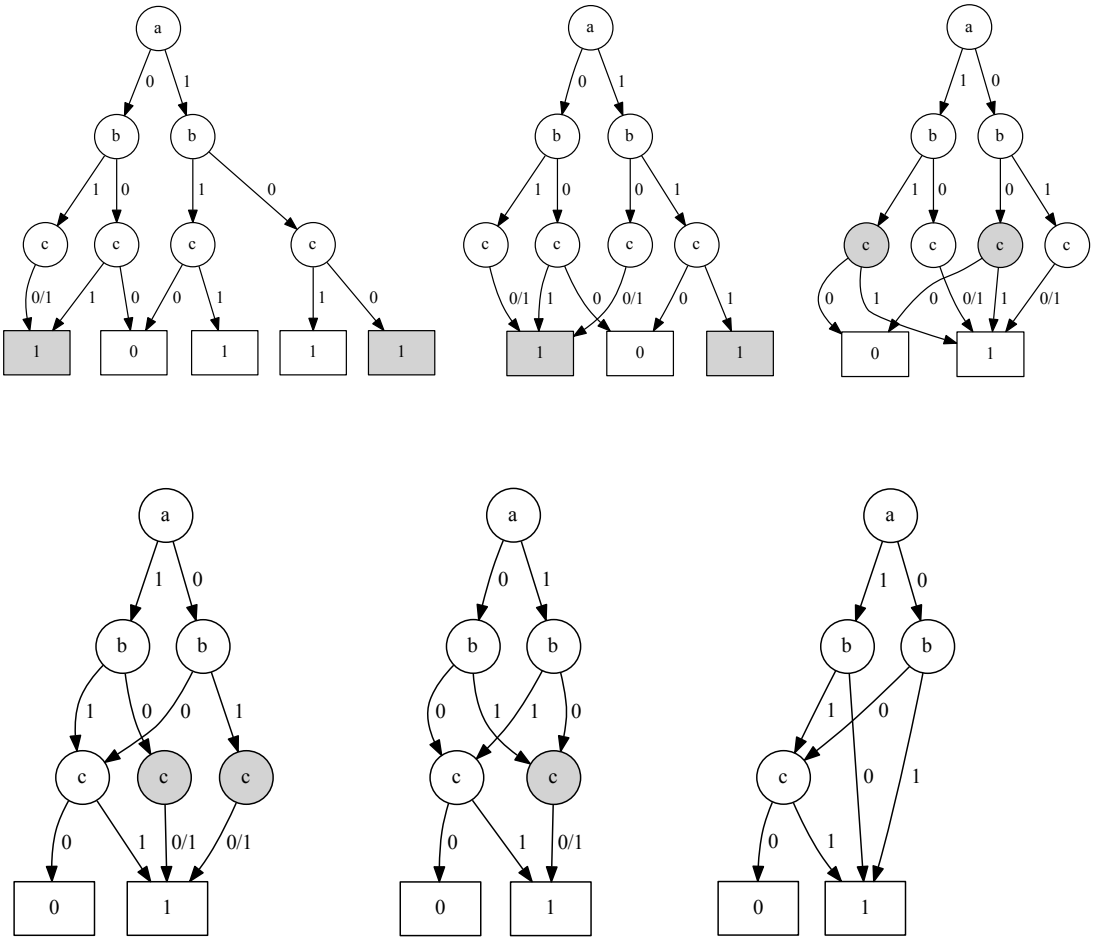


(a) Erzeugen Sie das zu f gehörende BDD.

/ 6

Spielen Sie auch mit diesem Skript für den XWizard herum.

```
bdd:
01111101
--declarations--
simplifySteps=-1
--declarations-end--
```



(b) Nennen Sie einen Vorteil von BDDs gegenüber der Darstellung durch Wahrheitstabellen.

/ 1

Lösung:

Vorteil: Reduzierte Darstellungsform von booleschen Funktionen

(c) Wofür werden BDDs heutzutage normalerweise eingesetzt? Begründen Sie Ihre Antwort kurz.

/ 1

Lösung:

Einsatzbereich: Datenstruktur für Hardwareentwurf und -optimierung, Schaltungsentwurf z.B. Datenstruktur bei CAD-Werkzeugen da geringerer / reduzierter Speicherbedarf

Aufgabe 8 **10 Punkte**

2015-N-08 **Zahlendarstellung**

/ 10

Gegeben sei die folgende Gleitkommadarstellung mit sechs Bits, aufgeteilt in ein Vorzeichenbit (VZ), Drei-Bit-Charakteristik (c_0, c_1, c_2) und Zwei-Bit-Mantisse (m_{-1}, m_{-2}) (**ohne** IEEE-typische Sonderfälle):

VZ	c_2	c_1	c_0	m_{-1}	m_{-2}

(a) Welche **Absolutbeträge** (nur positive Werte) können mit der gegebenen Aufteilung in Charakteristik und Mantisse minimal und maximal dargestellt werden?

/ 2

Lösung:

Drei Bit Charakteristik $\Rightarrow q = 2^{3-1} - 1 = 3$

Minimal darstellbarer Betrag: $(1 + 0 + 0) \cdot 2^{-3} = \frac{1}{8} = 0.125$

Maximal darstellbarer Betrag: $(1 + \frac{1}{2} + \frac{1}{4}) \cdot 2^4 = 1.75 \cdot 16 = 28$

(b) Es genügt, maximal drei Bits einer derart dargestellten Zahl zu verändern, um ihren Betrag zu halbieren. Nennen Sie diese Bits und erklären Sie kurz.

/ 2

Lösung:

Zur Halbierung der Zahl muss der Exponent über der Basis 2 um eins verringert werden. Dazu muss vom Wert der Charakteristik 1 abgezogen werden, man muss also die drei Bit der Charakteristik betrachten.

(c) Wandeln Sie die Dezimalzahl $(1,5625)_{10}$ **so genau wie möglich** in das angegebene Gleitkommaformat um (Vorzeichen, Drei-Bit-Charakteristik und Zwei-Bit-Mantisse).

/ 4

(1) Welcher Binärstring ergibt sich?

Lösung:

- VZ: 0
- Charakteristik: $(011)_2$
- Mantisse: $(10)_2$

$\Rightarrow (001110)_{GPZ}$

(2) Wie groß ist der Fehler bei der Umwandlung?

Lösung:

$(001110)_{GPZ} = (1 + \frac{1}{2}) \cdot 2^{3-3} = 1.5 \Rightarrow Fehler = 0.0625.$

(3) Welcher Teil der Gleitkommadarstellung müsste um wie viele Bits verlängert werden, um die Zahl exakt darzustellen?

Lösung:

$0.0625 = \frac{1}{16} = 2^{-4} \Rightarrow$ Mantisse um 2 Bit verlängern um 2^{-4} darzustellen.

(d) Betrachten Sie nun Gleitkommazahlen beliebiger Länge und Aufteilung. Welcher Zahlenbereich ist nicht darstellbar, wenn keine negativen Exponenten erlaubt sind?

Lösung:

Da die Mantisse m immer ≥ 1 ist, wäre der kleinste darstellbare Betrag $(1 + 0) \times 2^0 = 1$ und Zahlen aus dem Intervall $(-1, 1)$ könnten nicht dargestellt werden, egal wie sehr die Genauigkeit erhöht würde.

/ 2

Aufgabe 9 **8 Punkte**

2015-N-09 **Assembler und Pipelining**

/ 8

Die Befehle einer Assembler-Sprache seien folgendermaßen aufgebaut, wobei Q für Quelle steht und Z für Ziel:

```
OpCode Q1, (Q2,) Z
```

Für **unmittelbare Adressierung** wird das Präfix # verwendet, **direkte Adressierung** geben wir ohne Präfix an. `R1`, `R2` und `R3` seien zu Beginn mit 0 initialisiert, $m, n \in \mathbb{R}$.

Gegeben sei das folgende Assemblerprogramm:

```
I   | LOAD  #m  R1
II  | LOAD  #n  R2
III | ADD   R1  R2  R3
IV  | ADD   R3  R3  R3
```

Dem Assemblerprogramm liege folgender Maschinenbefehlszyklus zugrunde:

- *IF* (Instruction Fetch): Nächster Befehl wird gelesen.
- *ID* (Instruction Decode): OpCode wird decodiert.
- *EX* (Execute): Operanden werden gelesen und Operation wird ausgeführt.
- *WB* (Write Back): Ergebnis wird in Zielregister geschrieben.

(a) Wie ist der Inhalt von `R3` nach Ablauf des Programms?

Lösung:

`P` addiert Werte aus `R1` und `R2`, verdoppelt das Ergebnis und schreibt es in `R3`.
$R3 = 2(m + n)$.

/ 3

Der Maschinenbefehlszyklus wird nun in einer Pipeline parallelisiert. Jeder Schritt benötigt genau einen Takt und in jedem Takt wird ein neuer Befehl geholt (*IF*). Der `LOAD`-Befehl liest die Quelle während *EX* und schreibt das Zielregister während *WB*. Alles, was **während** *WB* geschrieben wird, steht erst einen Takt **nach** der *WB*-Stufe im Zielregister zum Lesen zur Verfügung.

(b) Geben Sie in der folgenden Tabelle die Registerinhalte in jedem Takt an, sowie die Nummern der Befehle, die sich gerade in den verschiedenen Pipelinestufen befinden. Warum kommt das Programm zu einem anderen Ergebnis als in Teilaufgabe (a)?

/ 4

Lösung:

Takt	*IF*	*ID*	*EX*	*WB*		R1	R2	R3
1	I					0	0	0
2	II	I						
3	III	II	I					
4	IV	III	II	I				
5		IV	III	II		*m*		
6			IV	III			*n*	
7				IV				*m*
8								0
9								
10								
11								

Sowohl Befehl `III` als auch Befehl `IV` greifen lesend auf ein Register zu, bevor der vorherige Befehl seinen Schreibzugriff darauf beendet hat (`III` auf `R2` in Takt 5 und `IV` auf `R3` in Takt 6) $\Rightarrow$ Zwischenergebnis steht noch nicht bereit, falsche Werte werden gelesen.

(c) Wie viele Takte werden für die Ausführung der Programme in den Aufgabenteilen (a) und (b) benötigt?

/ 1

Lösung:

Teilaufgabe (a): 16 Takte

Teilaufgabe (b): 7 Takte

Aufgabe 10	**8 Punkte**
2015-N-10	**Betriebssysteme**

/ 8

(a) Erläutern Sie jeweils kurz die folgenden Aspekte von Sicherheit in Betriebssystemen.

(1) **Zuverlässigkeit**

/ 1

Lösung:

Die Aufgaben eines Betriebssystems sollen korrekt ihrer entsprechenden Spezifikationen erfüllt werden.

(2) **Verfügbarkeit**

/ 1

Lösung:

Das Betriebssystem soll auch beim Ausfall einzelner Hard- und Softwarekomponenten dem Nutzer weiterhin zur Verfügung stehen.

(3) **Vertraulichkeit**

/ 1

Lösung:

Das Betriebssystem soll Dienste bereitstellen, um den Zugriff auf Informationen von Kommunikationspartnern und Benutzern einzuschränken.

(4) **Zugriffsschutz**

/ 1

Lösung:

Das Betriebssystem soll unberechtigten Zugriff auf Daten und Dienste verhindern.

(5) **Nachvollziehbarkeit**

/ 1

Lösung:

Das Betriebssystem muss gewährleisten, dass Zugriffe auf Daten oder Dienste dokumentiert und dadurch im Nachhinein nachvollziehbar sind.

(b) Welcher der Sicherheitsaspekte

Zuverlässigkeit,

Verfügbarkeit,

Vertraulichkeit,

Zugriffsschutz und

Nachvollziehbarkeit

wird **jeweils** in den folgenden beiden Situationen **am stärksten** berührt? Begründen Sie Ihre Antwort kurz.

/ 3

(1) Ein Geheimdienst hört ein Gespräch ab, das über eine unverschlüsselte Voice-over-IP-Verbindung über das Internet geführt wird.

Lösung:

Der Aspekt der Vertraulichkeit ist in dieser Situation berührt. Da das Gespräch unverschlüsselt über das Internet stattfindet, können grundsätzlich Dritte auf die übertragenen Daten zugreifen. (Zugriffsschutz kann bei der Übertragung über ein unsicheres System grundsätzlich nicht gewährleistet werden) Das System sollte einen kryptographischen Dienst bereitstellen, der ein Mithören trotz Mitschneiden des Gesprächs unmöglich macht.

(2) Ein fehlerhafter Webcam-Treiber führt dazu, dass das Betriebssystem mitten in einer Videokonferenz vollständig abstürzt.

Lösung:

Der Aspekt der Verfügbarkeit ist in dieser Situation berührt. Aufgrund des fehlerhaften Software-Treibers sollte lediglich die Videokonferenz und nicht das komplette Betriebssystem abstürzen.

Alternativ kann auch mit der Zuverlässigkeit argumentiert werden, da der Dienst während seiner Nutzung ausfällt und zum Absturz des Systems führt.

Aufgabe 11	**4 Punkte**
2015-N-11	**Dateiorganisation**

/ 4

Im Zuge der Entwicklung von immer größeren Festplatten gewinnen Dateiverwaltungssysteme immer mehr an Bedeutung.

(a) Nennen Sie vier wichtige Anforderungen an Dateiverwaltungssysteme.

/ 2

Lösung:

(1) Speicherung der Daten auf externen, großen, nichtflüchtigem Speichermedium
(2) Sicherheitskopien
(3) Übertragungsmöglichkeiten auf andere DV-Anlagen
(4) schnelle Verfügbarkeit
(5) schneller und sicherer Änderungsdienst
(6) Möglichkeit zur Strukturierung zusammengehörender Daten zu einer Einheit
(7) Sicherheitsaspekte wie Zugriffsrechte

(b) Nennen Sie vier Standardoperationen auf Dateien.

/ 2

Lösung:

(1) Lesen / Schreiben
(2) Einfügen / Hinzufügen
(3) Entfernen
(4) Suchen