Grundbegriffe der Informatik
Aufgabenblatt 3

Matr.nr.:	
Nachname:	
Vorname:	
Tutorium:	Nr.	Name des Tutors:

Ausgabe: 23. November 2016

Abgabe: 08. Dezember 2016, 16:00 Uhr
im GBI-Briefkasten im Untergeschoss
von Gebäude 50.34

Lösungen werden nur korrigiert, wenn sie

- rechtzeitig,
- in Ihrer eigenen Handschrift,
- mit dieser Seite als Deckblatt und
- in der oberen **linken** Ecke zusammengeheftet

abgegeben werden.

Vom Tutor auszufüllen:

erreichte Punkte

Blatt 3:	/ 44,5	(4 ECTS: 33,5)
Blätter 1 – 3:	/ 101,5	(4 ECTS: 90,5)

Mit **[nicht 4ECTS]** gekennzeichnete Aufgaben werden von Studenten, die den 4ECTS Schein der Vorlesung machen wollen, bitte nicht bearbeitet.

Aufgabe 3.1 (1 + 1 + 1 + 1= 4 Punkte)

a) Es sei $w = \texttt{11001}$. Geben Sie $u = \mathrm{Num}_2(w)$ und $v = \mathrm{Num}_3(w)$ an.

b) Geben Sie $\mu = \mathrm{Repr}_3(327)$ und $\nu = \mathrm{Repr}_9(327)$ an.

c) Das Wort μ der vorangegangenen Teilaufgabe hat die Länge 6. Geben Sie $\xi = \mathrm{Repr}_9(\mathrm{Num}_3(\mu(0)\mu(1))) \cdot \mathrm{Repr}_9(\mathrm{Num}_3(\mu(2)\mu(3))) \cdot \mathrm{Repr}_9(\mathrm{Num}_3(\mu(4)\mu(5)))$ und $\zeta = \mathrm{Num}_9(\xi)$ an.
Erinnerung: Für jedes $i \in \mathbb{Z}_6$ ist $\mu(i)$ das i-te Zeichen des Wortes μ.

d) Geben Sie $\mathrm{Num}_{11}(\mu)$ und $\mathrm{Num}_{13}(\nu)$ in Hexadezimaldarstellung an.

Lösung 3.1

a) $u = \mathrm{Num}_2(w) = 1 \cdot 2^0 + 1 \cdot 2^3 + 1 \cdot 2^4 = 1 + 8 + 16 = 25$
$v = \mathrm{Num}_3(w) = 1 \cdot 3^0 + 1 \cdot 3^3 + 1 \cdot 3^4 = 1 + 27 + 81 = 109$

b) $\mu = \texttt{110010}$
$\nu = \texttt{403}$

c) $\xi = \mathrm{Repr}_9(\mathrm{Num}_3(\mu(0)\mu(1))) \cdot \mathrm{Repr}_9(\mathrm{Num}_3(\mu(2)\mu(3))) \cdot \mathrm{Repr}_9(\mathrm{Num}_3(\mu(4)\mu(5))) =$
$\mathrm{Repr}_9(\mathrm{Num}_3(\mu(0)\mu(1))) \cdot \mathrm{Repr}_9(\mathrm{Num}_3(\mu(2)\mu(3))) \cdot \mathrm{Repr}_9(\mathrm{Num}_3(\mu(4)\mu(5)))$
$\xi = 403 = \nu$
$\zeta = 327$

d) $\mathrm{Num}_{11}(\mu) = 175.703$ somit $\mathrm{Repr}_{16}(175.703) = \texttt{2AE57}$. $\mathrm{Num}_{13}(\nu) = 679$ somit $\mathrm{Repr}_1(679) = \texttt{2A7}$

Aufgabe 3.2 (2 + 1 + 2 + 1 + 2 + 1 = 9 Punkte)

Sei $w = aabceefgeebdaabbceeffghdcbbeefbbbbghhie$

a) Konstruieren Sie den Huffman-Baum für w

b) Geben Sie an, welche Huffman-Codierung man für die in w vorkommenden Symbole aus dem Baum aus Teilaufgabe a) ablesen kann.

c) Konstruieren Sie einen zweiten Huffman-Baum für w, der sich vom Baum aus Teilaufgabe a) unterscheidet.

d) Geben Sie an, welche Huffman-Codierung man für die in w vorkommenden Symbole aus dem Baum aus Teilaufgabe c) ablesen kann.

e) Konstruieren Sie den Huffmann-Baum für ein Block-Code mit Blocklänge 4 für w.

f) Geben Sie an, welche Huffman-Codierung man für die in w vorkommenden Blöcke aus dem Baum aus Teilaufgabe e) ablesen kann.

Lösung 3.2

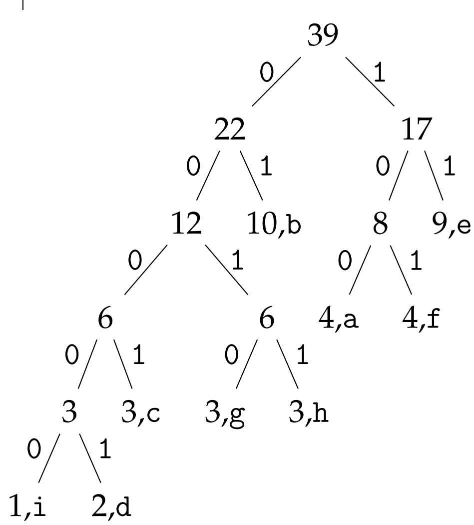
Häufigkeiten der Symbole in w:

i	1
d	2
c	3
g	3
h	3
a	4
f	4
e	9
b	10

Häufigkeiten der Viererblöcke in w:

aabc	1
eefg	1
eebd	1
aabb	1
ceef	1
fghd	1
cbbe	1
efbb	1
bbgh	1
hie	1

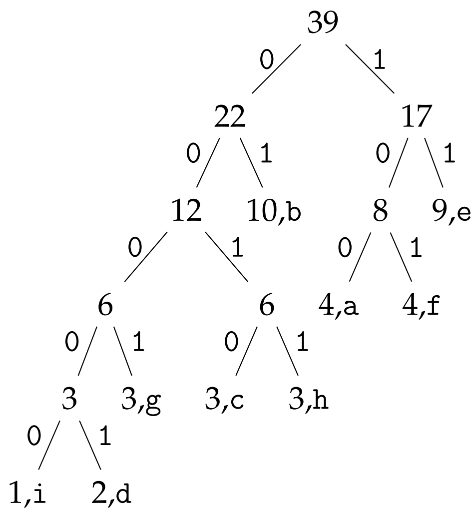
a)



b)

x	a	b	c	d	e	f	g	h	i
h(x)	100	01	0001	00001	11	101	0010	0011	00000

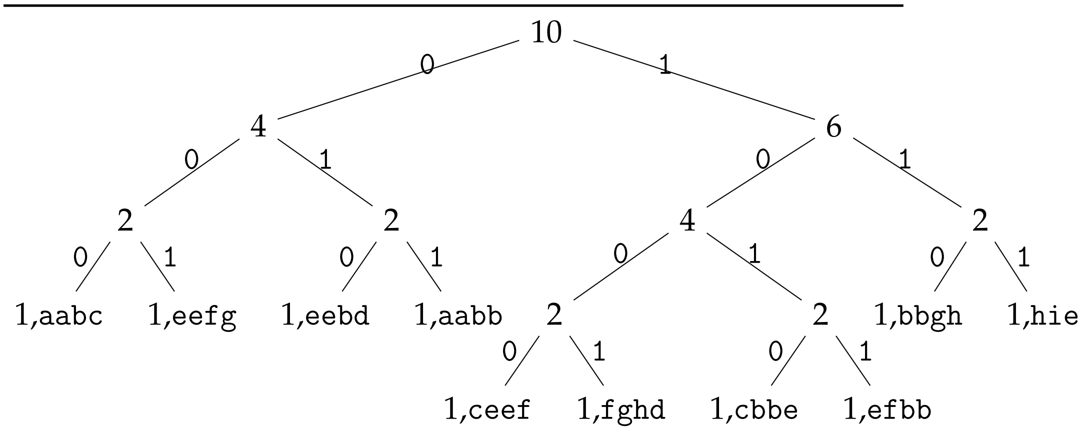
c)



d)

x	a	b	c	d	e	f	g	h	i
h(x)	100	01	0010	00001	11	101	0001	0011	00000

e)



f)

x	aabc	eefg	eebd	aabb	ceef	fghd	cbbe	efbb	bbgh	hie
h(x)	000	001	010	011	1000	1001	1010	1011	110	111

Aufgabe 3.3 (1 + 4 + 4 = 9 Punkte)

a) Seien $x \in \mathbb{Z}_{16}$ und $y \in \mathbb{Z}_{16}$. Berechnen Sie $x +_{16} y$ und $y -_{16} x$ für $x = 14$ und $y = 8$.

b) Basierend auf der Definition von $+_k$, geben Sie eine induktive Definition für die Multiplikation $\cdot_k$ in $\mathbb{Z}_k$ an, so dass für alle $x \in \mathbb{N}_0$ und $y \in \mathbb{N}_0$ gilt:

$$x \cdot_k y = (x \cdot y) \ \mathbf{mod} \ k$$

c) Beweisen Sie durch vollständige Induktion: Für ein $x \in \mathbb{N}_0$ ist für jedes $y \in \mathbb{N}_0$:

$$(x \cdot y) \ \mathbf{mod} \ k = (x \ \mathbf{mod} \ k) \cdot_k (y \ \mathbf{mod} \ k) \tag{1}$$

Lösung 3.3

a) $14 +_{16} 8 = (8 + 14) \ \mathbf{mod} \ 16 = 6$ und $8 -_{16} 14 = 10$

b)

$$x \cdot_k 0 = 0$$
$$\text{für jedes } n \in \mathbb{N}_0 \colon x \cdot_k (n+1) = x +_k (x \cdot_k n)$$

c) *Induktionsanfang*: Sei n=0. Dann gilt: $(x \cdot 0) \ \mathbf{mod}\ k = 0 \ \mathbf{mod}\ k = 0 = (x \ \mathbf{mod}\ k) \cdot_k 0 = (x \ \mathbf{mod}\ k) \cdot_k (0 \ \mathbf{mod}\ k)$
Induktionsschritt: Sei $n \in \mathbb{N}_0$ so, dass gilt: $(x \cdot n) \ \mathbf{mod}\ k = (x \ \mathbf{mod}\ k) \cdot_k (n \ \mathbf{mod}\ k)$. Dann gilt gemäß Definition in Teilaufgabe b) und induktiver Definition der Lösung von Teilaufgabe b):
$(x \cdot (n+1)) \ \mathbf{mod}\ k = (x \cdot n + x) \ \mathbf{mod}\ k = (((x \cdot n) \ \mathbf{mod}\ k) + (x \ \mathbf{mod}\ k)) \ \mathbf{mod}\ k = ((x \ \mathbf{mod}\ k) \cdot_k (n \ \mathbf{mod}\ k) \ \mathbf{mod}\ k) +_k (x \ \mathbf{mod}\ k) = ((x \ \mathbf{mod}\ k) \cdot_k (n \ \mathbf{mod}\ k)) +_k ((x \ \mathbf{mod}\ k)(1 \ \mathbf{mod}\ k)) = (x \ \mathbf{mod}\ k) \cdot_k ((n+1) \ \mathbf{mod}\ k)$

Aufgabe 3.4 (1 + 1 + 1 + 4 + 4= 11 Punkte) [nicht 4ECTS]
Es sei $\mathrm{Val} = \{\mathtt{0}, \mathtt{1}\}^8$, es sei $\mathrm{Adr} = \{\mathtt{0}, \mathtt{1}\}^{32}$ und es sei $\mathrm{Mem} = \mathrm{Val}^{\mathrm{Adr}}$. Die Addition modulo 2^8 zweier Zahlen in Binärdarstellung der Länge 8 ist gegeben durch die Abbildung

$$\begin{aligned} \mathrm{add}_{\mathrm{Val}} \colon & \mathrm{Val} \times \mathrm{Val} \to \mathrm{Val}, \\ & (u, v) \mapsto \mathrm{bin}_8((\mathrm{Num}_2(u) + \mathrm{Num}_2(v)) \ \mathbf{mod}\ 2^8), \end{aligned}$$

und die Addition modulo 2^{32} zweier Zahlen in Binärdarstellung der Länge 32 beziehungsweise 8 ist gegeben durch die Abbildung

$$\begin{aligned} \mathrm{add}_{\mathrm{Adr}} \colon & \mathrm{Adr} \times \mathrm{Val} \to \mathrm{Adr}, \\ & (a, v) \mapsto \mathrm{bin}_{32}((\mathrm{Num}_2(a) + \mathrm{Num}_2(v)) \ \mathbf{mod}\ 2^{32}). \end{aligned}$$

Ein (Warte-)Schlange ist eine Datenstruktur mit drei grundlegenden Operationen:

a) „enqueue“ legt einen Wert in die Schlange;
b) „dequeue“ nimmt den ältesten Wert aus der Schlange;
c) „first“ liefert den ältesten Wert, ohne ihn aus der Schlange zu nehmen.

In unserem Speichermodell kann ein Schlange mit höchstens 2^8-vielen Werten durch eine Adresse repräsentiert werden. Diese Addresse sowie das nächste Feld speichern eine Zeiger auf den Anfang und das Ende der Schlange. Der Zeiger gibt die Adresse relative zur Grundaddresse +2 an. Die Abbildungen init_queue, is_empty, enqueue, dequeue und first bilden eine Schnittstelle zur Verwaltung von Schlangen und sind gegeben durch:

$$\begin{aligned} \mathrm{init\_queue} \colon & \mathrm{Mem} \times \mathrm{Adr} \to \mathrm{Mem}, \\ & (m, a) \mapsto \mathrm{memwrite}(\mathrm{memwrite}(m, a, \mathrm{bin}_8(0)), \mathrm{add}_{\mathrm{Adr}}(a, 1), \mathrm{bin}_8(0)), \end{aligned}$$

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

$$(m, a) \mapsto \begin{cases} \mathbf{w}, & \text{falls } \mathrm{memread}(m, a) = \mathrm{memread}(m, \mathrm{add}_{\mathrm{Adr}}(a, 1)), \\ \mathbf{f}, & \text{sonst,} \end{cases}$$

$$\begin{aligned} \mathrm{enqueue}\colon \mathrm{Mem} \times \mathrm{Adr} \times \mathrm{Val} &\to \mathrm{Mem}, \\ (m, a, v) &\mapsto \mathrm{memwrite}(m', a', \mathrm{add}_{\mathrm{Val}}(\mathrm{memread}(m', a'), \mathrm{bin}_8(1))), \\ &\qquad \text{wobei } m' = \mathrm{memwrite}(m, \mathrm{add}_{\mathrm{Adr}}(a^*, \mathrm{memread}(m, a')), v), \\ &\qquad a' = \mathrm{add}_{\mathrm{Adr}}(a, 1), \\ &\qquad a^* = \mathrm{add}_{\mathrm{Adr}}(a, \mathrm{bin}_8(2)) \end{aligned}$$

$$\begin{aligned} \mathrm{dequeue}\colon \mathrm{Mem} \times \mathrm{Adr} &\to \mathrm{Mem}, \\ (m, a) &\mapsto \begin{cases} m, & \text{falls } \mathrm{is\_empty}(m, a), \\ \mathrm{memwrite}(m, a, \mathrm{add}_{\mathrm{Val}}(\mathrm{memread}(m, a), \mathrm{bin}_8(1))), & \text{sonst.} \end{cases} \end{aligned}$$

$$\begin{aligned} \mathrm{first}\colon \mathrm{Mem} \times \mathrm{Adr} &\to \mathrm{Val}, \\ (m, a) &\mapsto \mathrm{memread}(m, \mathrm{add}_{\mathrm{Adr}}(\mathrm{add}_{\mathrm{Adr}}(a, \mathrm{memread}(m, a)), bin_8(2))) \end{aligned}$$

Für jeden Speicher $m \in \mathrm{Mem}$, jede Adresse $a \in \mathrm{Adr}$ und jeden Wert $v \in \mathrm{Val}$, initialisiert $\mathrm{init\_queue}(m, a)$ einen Schlange bei a in m, prüft $\mathrm{is\_empty}(m, a)$, ob die Schlange bei a in m leer ist oder nicht, legt $\mathrm{enqueue}(m, a, v)$ den Wert v auf die Schlange bei a in m, nimmt $\mathrm{dequeue}(m, a)$ den ältesten Wert von der Schlange bei a in m und liefert $\mathrm{first}(m, a)$ den ältesten Wert von der Schlange bei a in m.

a) Es sei $m \in \mathrm{Mem}$ und es sei $a = \mathrm{bin}_{32}(0)$. Geben Sie den Wert

$$\mathrm{first}(\mathrm{dequeue}(\mathrm{enqueue}(\mathrm{enqueue}(\mathrm{init\_queue}(m, a), a, \texttt{00101111}), a, \texttt{00001100}), a), a)$$

an.

b) Es sei $m \in \mathrm{Mem}$, es sei $a = \mathrm{bin}_{32}(0)$ und es sei

$$m' = \mathrm{enqueue}(\mathrm{enqueue}(\mathrm{init\_queue}(m, a), a, \texttt{11111111}), a, \texttt{00000001}).$$

Geben Sie den Wert $\mathrm{add}_{\mathrm{Val}}(\mathrm{first}(m', a), \mathrm{first}(\mathrm{dequeue}(m', a), a))$ an.

c) Geben Sie den Wert der Speicheradresse an, den a maximal haben darf, so daß die von der Schlange addressierten Speichezellen nie zu einem Überlauf in den verfügbaren Speicheraddressen führen.

d) Definieren Sie induktiv, unter ausschließlicher Verwendung der Abbildungen $\mathrm{add}_{\mathrm{Val}}$, is_empty, dequeue und first, eine Abbildung $\mathrm{sum}\colon \mathrm{Mem} \times \mathrm{Adr} \to \mathrm{Val}$ derart, dass für jeden Speicher $m \in \mathrm{Mem}$ und jede Adresse $a \in \mathrm{Adr}$ gilt, dass $\mathrm{sum}(m, a)$ die Binärdarstellung der Summe modulo 2^8 aller Werte, interpretiert als Binärdarstellungen von Zahlen, in der Schlange bei a in m ist, wobei die leere Summe per Definition 0 ist.

e) Eine Prioritäten-Schlange ist eine Erweiterung der Schlange. Beim Einfügen kann man zusätzliche die Priorität 1 oder 0 angeben. Beim Ausgeben werden zunächst alle Element ausgegeben, die Priorität 1 haben und danach die Element mit Priorität 0. Definieren Sie die Schnittstelle für die Prioritätenschlange durch die Verwendung von zwei Schlangen:

$$\begin{aligned}\text{init\_queue}\colon \text{Mem} \times \text{Adr} &\to \text{Mem},\\ (m,a) &\mapsto ?\end{aligned}$$

$$\begin{aligned}\text{is\_empty}\colon \text{Mem} \times \text{Adr} &\to \mathbb{B},\\ (m,a) &\mapsto ?\end{aligned}$$

$$\begin{aligned}\text{enqueue}\colon \text{Mem} \times \text{Adr} \times \text{Val} \times \mathbb{B} &\to \text{Mem},\\ (m,a,v,p) &\mapsto ?\end{aligned}$$

$$\begin{aligned}\text{dequeue}\colon \text{Mem} \times \text{Adr} &\to \text{Mem},\\ (m,a) &\mapsto ?\end{aligned}$$

$$\begin{aligned}\text{first}\colon \text{Mem} \times \text{Adr} &\to \text{Val},\\ (m,a) &\mapsto ?\end{aligned}$$

Lösung 3.4

a) `00001100`

b) `00000000`

c) $a_{max} = 2^{32} - 2^8 - 2 = 429.4967.038$ oder in Hexadezimal: $a_{max} =$ `FFFFFEFE`

d)

$$\begin{aligned}\text{sum}\colon \text{Mem} \times \text{Adr} &\to \text{Val},\\ (m,a) &\mapsto \begin{cases} 0, & \text{falls is\_empty}(m,a) = \mathbf{w},\\ \text{add}_{\text{Val}}(\text{sum}(\text{dequeue}(m,a),a), \text{first}(m,a)), & \text{sonst.}\end{cases}\end{aligned}$$

e)

$$\begin{aligned}\text{init\_queue}\colon \text{Mem} \times \text{Adr} &\to \text{Mem},\\ (m,a) &\mapsto \text{init\_queue}(\text{init\_queue}(m,a), \text{add}_{\text{Adr}}(a, 2^8+2))\end{aligned}$$

$$\text{is\_empty}\colon \text{Mem} \times \text{Adr} \to \mathbb{B},$$
$$(m,a) \mapsto \begin{cases} \text{is\_empty}(m, \text{add}_{\text{Adr}}(a, 2^8+2)), & \text{falls is\_empty}(m,a) = \mathbf{w}, \\ \text{is\_empty}(m,a), & \text{sonst.} \end{cases}$$

$$\text{enqueue}\colon \text{Mem} \times \text{Adr} \times \text{Val} \times \mathbb{B} \to \text{Mem},$$
$$(m,a,v,b) \mapsto \begin{cases} \text{enqueue}(m, \text{add}_{\text{Adr}}(a, 2^8+2), v), & \text{falls } b = \mathbf{w}, \\ \text{enqueue}(m,a,v), & \text{sonst.} \end{cases}$$

$$\text{dequeue}\colon$$
$$\text{Mem} \times \text{Adr} \to \text{Mem},$$
$$(m,a) \mapsto \begin{cases} \text{dequeue}(m,a), & \text{falls is\_empty}(m, \text{add}_{\text{Adr}}(a, 2^8+2)) = \mathbf{w}, \\ \text{dequeue}(m, \text{add}_{\text{Adr}}(a, 2^8+2)), & \text{sonst.} \end{cases}$$

$$\text{first}\colon$$
$$\text{Mem} \times \text{Adr} \to \text{Val},$$
$$(m,a) \mapsto \begin{cases} \text{first}(m,a), & \text{falls is\_empty}(m, \text{add}_{\text{Adr}}(a, 2^8+2)) = \mathbf{w}, \\ \text{first}(m, \text{add}_{\text{Adr}}(a, 2^8+2), v), & \text{sonst.} \end{cases}$$

Aufgabe 3.5 (5 Punkte)
Zeigen Sie, dass zu jeder aussagenlogischer Formel (F) eine äquivalente Formel existiert, welche nur die aussagenlogische Konnektive $\neg$ und $\wedge$ enthält.

Lösung 3.5

- Formeln dürfen nur die Konnektive $\neg$ (Negierung) und $\wedge$ (und) enthalten.
- Beide sind über 2 Inputs definiert:
- Zwei Formeln sind äquivalent wenn all Interpetationen äquivalent sind

P	Q	$P \vee Q$	$\neg P$	$\neg Q$	$\neg P \wedge \neg Q$	$\neg(\neg P \wedge \neg Q)$
0	0	0	1	1	1	0
0	1	1	1	0	0	1
1	0	1	0	1	0	1
1	1	1	0	0	0	1

- $\longrightarrow P \vee Q \equiv \neg(\neg P \wedge \neg Q)$

P	Q	$P \to Q$	$\neg Q$	$\neg Q \land P$	$\neg(\neg Q \land P)$
0	0	1	1	0	1
0	1	1	0	0	1
1	0	0	1	1	0
1	1	1	0	0	1

- $\longrightarrow P \to Q \equiv \neg(\neg Q \land P)$
- Jetzt können wir folgende Funktionen definieren:
- $\hat{f}_\lor : A^*_{AL} \times A^*_{AL} \to A^*_{AL} : (G, H) \mapsto (f_\neg(f_\land(f_\neg(G), f_\neg(H))))$
- $\hat{f}_\to : A^*_{AL} \times A^*_{AL} \to A^*_{AL} : (G, H) \mapsto (f_\neg(f_\land(f_\neg(H), G)))$
- Definiere $\hat{\text{For}}_{AL} \subset \text{For}_{AL}$: parallel zu For_{AL} mit:
  - $\hat{f}_\lor$ statt $f_\lor$
  - $\hat{f}_\to$ statt $f_\to$
- Nach definition enthält die Untermenge $\hat{\text{For}}_{AL}$ nur die Konnektive $\neg$ und $\land$ und es existiert für jede Formel in For_{AL} eine äquivalente Formel in $\hat{\text{For}}_{AL}$.

Aufgabe 3.6 (1,5+2+3=6,5 Punkte)

In einem Kloster leben zwei Arten von Schwestern. Die Schwestern des Lichts, die nie lügen, und die Schwestern der Dunkelheit, die immer lügen aber schlau sind, so dass es nicht sofort erkennbar ist. Bei einem Besuch im Kloster treffen Sie auf drei Schwestern X, Y, Z.

- X sagt folgendes: Y und Z sagen genau dann die Wahrheit, wenn Z die Wahrheit sagt.
- Y sagt: Wenn X und Z die Wahrheit sagen, dann ist es nicht der Fall, dass X die Wahrheit sagt, wenn Y und Z die Wahrheit sagen.
- Z sagt: Y lügt genau dann, wenn X oder Z die Wahrheit sagen.

a) Formulieren Sie diese Ausagen als Aussagenlogische Formeln A_X, A_Y, A_Z. Verwenden Sie dazu W_I (Schwester I sagt die Wahrheit) als Variabeln.

b) Bilden Sie aus allen drei Aussagen eine aussagenlogische Formel. Ist diese Formel erfüllbar? Wenn ja, dann geben Sie bitte eine entsprechende Belegung an.

c) Welche der drei Schwestern sind Schwestern der Dunkelheit?

Lösung 3.6

a)
- $A_X := (W_Y \land W_Z) \leftrightarrow W_Z$
- $A_Y := (W_X \land W_Z) \to \neg((W_Y \land W_Z) \to W_X)$
- $A_Z := (W_X \lor W_Z) \leftrightarrow \neg W_Y$

b) $F := (A_X \leftrightarrow W_X) \land (A_Y \leftrightarrow W_Y) \land (A_Z \leftrightarrow W_Z)$ Sei: $T_I := A_I \leftrightarrow W_I$

W_X	W_Y	W_Z	A_X	A_Y	A_Z	T_X	T_Y	T_Z	F
0	0	0	1	1	0	0	0	1	0
0	0	1	0	1	0	1	0	0	0
0	1	0	1	1	0	0	1	1	0
0	1	1	1	1	0	0	1	0	0
1	0	0	1	1	1	1	0	0	0
1	0	1	0	0	1	0	1	1	0
1	1	0	1	1	0	1	1	1	1
1	1	1	1	0	0	1	0	0	0

Die Formel ist erfüllbar mit der Belegung $W_X = 1$, $W_Y = 1$ und $W_Z = 0$

c) Schwester Z ist deswegen eine Schwester der Dunkelheit