1. Übungsblatt (mit Lösungen)

3.0 VU Formale Modellierung SS2021

Marion Scholz, Gernot Salzer

13. Mai 2021

Aufgabe 1 (3 Punkte)

Geben Sie für die folgenden Schlussfolgerungen die zugrundeliegende Inferenzregel an und stellen Sie fest, ob diese gültig ist. Wenn ja, geben Sie unter Verwendung von Alltagsbegriffen eine weitere Schlussfolgerung an, die derselben Regel folgt. Wenn nein, geben Sie ein (weiteres) Gegenbeispiel an, das heißt, geben Sie eine Schlussfolgerung an, die derselben Regel folgt und bei der die Prämissen wahr, die Schlussfolgerung aber falsch ist.

(a) Siehe Abbildung rechts.

(b) Wenn alle Tober xetig sind und Hituch ein Tober ist, dann ist Hituch xetig.

(c) Alle Katzen miauen. Bello ist keine Katze. Also miaut Bello nicht.

https://de.toonpool.com/artists/Erl_64

Lösung

(a)
$$\begin{array}{l}\text{Manche Menschen sind kriminell.}\\ \underline{\text{Alle Asylanten sind Menschen.}}\\ \text{Alle Asylanten sind kriminell.}\end{array} \qquad \text{Inferenzregel: } \begin{array}{l}\text{Manche } x \text{ sind } y.\\ \underline{\text{Alle } z \text{ sind } x.}\\ \text{Alle } z \text{ sind } y.\end{array}$$

Diese Inferenzregel ist **nicht gültig**. Ein weiteres Gegenbeispiel ist etwa:

$$\begin{array}{l}\text{Manche Tiere sind Säugetiere.}\\ \underline{\text{Alle Fliegen sind Tiere.}}\\ \text{Alle Fliegen sind Säugetiere.}\end{array}$$

(b) Auch ohne die Bedeutung der Worte „Tober“, „xetig“ oder „Hituch“ zu kennen, lässt sich die Inferenzregel und ihre Gültigkeit analysieren.

$$\begin{array}{l}\text{Alle Tober sind xetig.}\\ \underline{\text{Hituch ist ein Tober.}}\\ \text{Hituch ist xetig.}\end{array} \qquad \text{Inferenzregel: } \begin{array}{l}\text{Alle } x \text{ sind } y.\\ \underline{z \text{ ist ein } x.}\\ z \text{ ist } y.\end{array}$$

Diese Inferenzregel ist **gültig**. Andere Schlussfolgerung mit derselben Inferenzregel:

Alle Türme sind hoch.
Der Donauturm ist ein Turm.

Der Donauturm ist hoch.

(c) Alle Katzen miauen.
Bello ist keine Katze.

Also miaut Bello nicht.

Inferenzregel: Alle x können y.
z ist kein x.
---
z kann nicht y.

Diese Inferenzregel ist **nicht gültig**. Ein Gegenbeispiel ist etwa:

Alle Boote schwimmen
Der Flamingo ist kein Boot.

Der Flamingo kann nicht schwimmen.

Aufgabe 2 (4 Punkte)

Analysieren Sie den folgenden Text und identifizieren Sie die logische Struktur sowie die Elementaraussagen. Betrachten Sie dabei jeden Satz einzeln.

> Liebe Kundin, lieber Kunde!
> Wir gratulieren zum Erwerb dieses Produkts! Legen Sie Hammer und Schraubenzieher oder Akkuschrauber bereit. Packen Sie die Plastikteile aus, aber beschädigen Sie dabei die Beschichtung nicht. Nur wenn Sie die Version 2.0 des Modells erworben haben, stecken Sie nun alle Teile zusammen. Falls nicht, müssen Sie alle Teile verleimen. Falls Sie beim Aufbau Probleme haben oder Teile fehlen, wenden Sie sich bitte an unsere Service-Hotline.
> Gutes Gelingen!

Lösung

(a) *Liebe Kundin, lieber Kunde!*

Hierbei handelt es sich um keine Aussage, da diese Wortfolge weder wahr noch falsch sein kann.

(b) *Wir gratulieren zum Erwerb dieses Produkts!*

A ... Wir gratulieren zum Erwerb dieses Produkts!

Struktur: A
Formel: A

(c) *Legen Sie Hammer und Schraubenzieher oder Akkuschrauber bereit.*

Ein Befehl bzw. eine Aufforderung ist genau genommen keine Aussage. Will man die Struktur der Aufforderung analysieren und daraus Schlussfolgerungen ziehen, lässt sich der Satz aber in Aussagen umformen.

A ... Sie legen einen Hammer bereit.
B ... Sie legen einen Schraubenzieher bereit.
C ... Sie legen einen Akkuschrauber bereit.

Struktur: A und B oder C
Formel: $A \wedge (B \vee C)$ oder $A \wedge (B \not\equiv C)$

(d) *Packen Sie die Plastikteile aus, aber beschädigen Sie dabei die Beschichtung nicht.*

A ... Sie packen die Plastikteile aus.
B ... Sie beschädigen die Beschichtung.

Struktur: A und nicht B
Formel: $A \wedge \neg B$

(e) *Nur wenn Sie die Version 2.0 des Modells erworben haben, stecken Sie nun alle Teile zusammen.*

A ... Sie haben die Version 2.0 des Modells erworben.
B ... Sie stecken alle Teile zusammen.

Struktur: A genau dann, wenn B
Formel: $A \equiv B$ oder $B \equiv A$

(f) *Falls nicht, müssen Sie alle Teile verleimen.*

A ... Sie haben die Version 2.0 des Modells erworben.
B ... Sie müssen alle Teile verleimen.

Struktur: Wenn nicht A dann B.
Formel: $\neg A \supset B$

(g) *Falls Sie beim Aufbau Probleme haben oder Teile fehlen, wenden Sie sich bitte an unsere Service-Hotline.*

A ... Sie haben beim Aufbau Probleme.
B ... Es fehlen Teile.
C ... Sie wenden sich an die Service-Hotline.

Struktur: Wenn A oder B dann C.
Formel: $(A \vee B) \supset C$

(h) *Gutes Gelingen!*
Hierbei handelt es sich um keine Aussage, da „Gutes Gelingen!“ weder wahr noch falsch sein kann.

Aufgabe 3 (4 Punkte)

Maria, Clara und Peter überlegen, ob sie mit Tanja ins Kino mitkommen. Wie können die folgenden Sätze mit aussagenlogischen Formeln formalisiert werden?

(a) Maria kommt mit.

(b) Peter kommt vielleicht mit.

(c) Clara kommt mit, Peter aber nicht.

(d) Wenn Maria mitkommt, dann kommt Peter nicht mit.

(e) Nur wenn Maria mitkommt und Peter nicht, dann kommt Clara.

(f) Wenn Peter nicht kommt, dann kommt Clara nicht, und wenn Clara nicht kommt, dann kommt Peter nicht.

(g) Mindestens eine(r) der drei kommt mit.

(h) Genau zwei kommen mit.

(i) Höchstens zwei kommen mit.

Lösung

M ... Maria kommt mit.
P ... Peter kommt mit.
C ... Clara kommt mit.

(a) M

(b) $P \not\equiv \neg P$ oder $P \vee \neg P$ oder $\top$
Diese Aussage definiert keine Einschränkung.

(c) $C \wedge \neg P$

(d) $M \supset \neg P$

(e) $C \supset (M \wedge \neg P)$

(f) $\neg P \equiv \neg C$

(g) $M \vee P \vee C$

(h) $(M \wedge P \wedge \neg C) \vee (M \wedge \neg P \wedge C) \vee (\neg M \wedge P \wedge C)$

(i) $\neg(M \wedge P \wedge C)$ oder $\neg M \vee \neg P \vee \neg C$

Aufgabe 4 (4 Punkte)

(a) Zeigen Sie, dass die Menge $\{\mathsf{if}, \mathsf{xor}\}$ vollständig ist für die Klasse der aussagenlogischen Funktionen.

(b) Zeigen Sie, dass die Menge $\{\mathsf{iff}, \mathsf{or}\}$ nicht vollständig ist.

Anmerkung: Die Begründung, dass in jedem Ausdruck immer mindestens eine Variable vorkommen muss und es daher nicht möglich ist, nullstellige Funktionen (also Konstanten wie false) darzustellen, ist nicht ausreichend, da das nur ein Problem der gewählten Darstellung ist. Wenn Sie aber zeigen können, dass beispielsweise die einstellige konstante Funktion definiert durch $\mathsf{false}(x) = 0$ nicht darstellbar ist, ist das schlüssig.

Lösung

(a) Wir wissen aus der Vorlesung, dass die Funktionsmenge $\{\mathsf{and}, \mathsf{not}\}$ vollständig ist. Es reicht daher zu zeigen, dass diese beiden Funktionen durch if und xor darstellbar sind. Betrachten wir die Definition dieser Funktionen.

if	0	1
0	1	0
1	1	1

xor	0	1
0	0	1
1	1	0

Wir erhalten aus beiden Funktionen die Negation, wenn wir ein Argument fix auf 0 bzw. 1 setzen können:

$$\mathsf{not}\, x = (0 \;\mathsf{if}\; x) = (1 \;\mathsf{xor}\; x)$$

Die Konstanten 0 bzw. 1 erhalten wir, indem wir die Argumente der beiden Funktionen identisch wählen, da die Diagonale der Tabellen identisch 0 bzw. 1 ist. Somit lässt sich die Negation folgendermaßen durch if und xor ausdrücken.

$$\mathsf{not}\, x = ((x \;\mathsf{xor}\; x) \;\mathsf{if}\; x) = ((x \;\mathsf{if}\; x) \;\mathsf{xor}\; x)$$

Bei der Darstellung der Funktion and lassen wir uns von dem Wissen leiten, dass die Implikation eine Disjunktion ist, bei der ein Argument negiert ist.

$$x \;\mathsf{and}\; y = \mathsf{not}(\mathsf{not}\, x \;\mathsf{or}\; \mathsf{not}\, y) = \mathsf{not}(\mathsf{not}\, x \;\mathsf{if}\; y) \qquad \text{da } A \;\mathsf{if}\; B = A \;\mathsf{or}\; \mathsf{not}\, B$$

Damit sind wir fertig, da wir bereits gezeigt haben, dass sich not durch if und xor darstellen lässt. Wir können die Ausdrücke ineinander einsetzen, was folgendes Monster liefert.

$$\begin{aligned} x \;\mathsf{and}\; y &= \mathsf{not}(\mathsf{not}\, x \;\mathsf{if}\; y) = \mathsf{not}(((x \;\mathsf{xor}\; x) \;\mathsf{if}\; x) \;\mathsf{if}\; y) \\ &= ((((x \;\mathsf{xor}\; x) \;\mathsf{if}\; x) \;\mathsf{if}\; y) \;\mathsf{xor}\; (((x \;\mathsf{xor}\; x) \;\mathsf{if}\; x) \;\mathsf{if}\; y)) \;\mathsf{if}\; (((x \;\mathsf{xor}\; x) \;\mathsf{if}\; x) \;\mathsf{if}\; y) \end{aligned}$$

(b) Es genügt von einer einzigen Funktion zu zeigen, dass sie nicht durch iff und or darstellbar ist. Wir untersuchen, welche einstelligen Funktionen darstellbar sind. Da $\mathsf{iff}(x, x) = \mathsf{true}(x)$ gilt, berechnen wir iff und or für die Argumente x und $\mathsf{true}(x)$.

$$\begin{aligned} x \;\mathsf{iff}\; x &= \mathsf{true}(x) & x \;\mathsf{or}\; x &= x \\ \mathsf{true}(x) \;\mathsf{iff}\; x &= x & \mathsf{true}(x) \;\mathsf{or}\; x &= \mathsf{true}(x) \\ x \;\mathsf{iff}\; \mathsf{true}(x) &= x & x \;\mathsf{or}\; \mathsf{true}(x) &= \mathsf{true}(x) \\ \mathsf{true}(x) \;\mathsf{iff}\; \mathsf{true}(x) &= \mathsf{true}(x) & \mathsf{true}(x) \;\mathsf{or}\; \mathsf{true}(x) &= \mathsf{true}(x) \end{aligned}$$

Somit sind durch iff und or nur die identische Funktion x und die konstante Funktion $\mathsf{true}(x)$ ausdrückbar, nicht aber $\mathsf{not}(x)$ und $\mathsf{false}(x)$. Die Menge $\{\mathsf{iff}, \mathsf{or}\}$ ist daher nicht funktional vollständig.

Aufgabe 5 (3 Punkte)

Sei F die Formel $(((A \vee \neg B) \supset C) \equiv ((B \supset C) \vee \neg A))$.

(a) Zeigen Sie, dass F syntaktisch korrekt ist.

(b) Berechnen Sie schrittweise $\mathsf{val}_I(F)$ für $I(A) = 1$, $I(B) = 0$ und $I(C) = 0$.

(c) Verwenden Sie eine Wahrheitstafel um festzustellen, ob die Formel F gültig, erfüllbar, widerlegbar und/oder unerfüllbar ist.

Lösung

(a) Laut Vorlesung ist die Menge $\mathcal{A}$ der aussagenlogischen Formeln die kleinste Menge, für die gilt:

(a1) $\mathcal{V} \subseteq \mathcal{A}$

(a2) $\{\top, \bot\} \subseteq \mathcal{A}$

(a3) $\neg F \in \mathcal{A}$, wenn $F \in \mathcal{A}$.

(a4) $(F * G) \in \mathcal{A}$, wenn $F, G \in \mathcal{A}$ und $* \in \{\wedge, \uparrow, \vee, \downarrow, \equiv, \not\equiv, \supset, \subset\}$.

wobei $\mathcal{V} = \{A, B, C, \dots\}$ die Menge der aussagenlogischen Variablen ist.

Wir zeigen, dass $(((A \vee \neg B) \supset C) \equiv ((B \supset C) \vee \neg A))$ eine aussagenlogische Formel gemäß dieser Definition ist.

(1) Die Variablen A, B und C sind Formeln (a1).

(2) Da B eine Formel ist (Punkt 1), ist auch $\neg B$ eine Formel (a3).

(3) Da A und $\neg B$ Formeln sind (Punkt 1 bzw. 2), ist auch $(A \vee \neg B)$ eine Formel (a4).

(4) Da $(A \vee \neg B)$ und C Formeln sind (Punkt 3 bzw. 1), ist auch $((A \vee \neg B) \supset C)$ eine Formel (a4).

(5) Da B und C Formeln sind (Punkt 1), ist auch $(B \supset C)$ eine Formel (a4).

(6) Da A eine Formel ist (Punkt 1), ist auch $\neg A$ eine Formel (a3).

(7) Da $(B \supset C)$ und $\neg A$ Formeln sind (Punkt 5 bzw. 6), ist auch $((B \supset C) \vee \neg A)$ eine Formel (a4).

(8) Da $((A \vee \neg B) \supset C)$ und $((B \supset C) \vee \neg A)$ Formeln sind (Punkt 4 bzw. 7), ist auch $(((A \vee \neg B) \supset C) \equiv ((B \supset C) \vee \neg A))$ eine Formel (a4).

Dieselbe Argumentation in Form eines Baumes:

$$\dfrac{\dfrac{\dfrac{\dfrac{A \in \mathcal{V}}{A \in \mathcal{A}}\,a1 \quad \dfrac{\dfrac{B \in \mathcal{V}}{B \in \mathcal{A}}\,a1}{\neg B}\,a3}{(A \vee \neg B) \in \mathcal{A}}\,a4 \quad \dfrac{C \in \mathcal{V}}{C \in \mathcal{A}}\,a1}{((A \vee \neg B) \supset C) \in \mathcal{A}}\,a4 \quad \dfrac{\dfrac{\dfrac{B \in \mathcal{V}}{B \in \mathcal{A}}\,a1 \quad \dfrac{C \in \mathcal{V}}{C \in \mathcal{A}}\,a1}{(B \supset C) \in \mathcal{A}}\,a4 \quad \dfrac{\dfrac{A \in \mathcal{V}}{A \in \mathcal{A}}\,a1}{\neg A}\,a3}{((B \supset C) \vee \neg A) \in \mathcal{A}}\,a4}{(((A \vee \neg B) \supset C) \equiv ((B \supset C) \vee \neg A)) \in \mathcal{A}}\,a4$$

Die horizontalen Linien sind als wenn-dann zu lesen, wobei die Prämissen des Schlusses oberhalb und die Konklusion unterhalb der Linie angegeben werden. Neben dem Strich steht die angewendete Regel.

(b) $\mathsf{val}_I\,(((A \vee \neg B) \supset C) \equiv ((B \supset C) \vee \neg A))$
$= \mathsf{val}_I\,((A \vee \neg B) \supset C)$ iff $\mathsf{val}_I\,((B \supset C) \vee \neg A)$
$= (\mathsf{val}_I(A \vee \neg B)$ implies $\mathsf{val}_I(C))$ iff $(\mathsf{val}_I(B \supset C) \vee \mathsf{val}_I(\neg A))$
$= ((\mathsf{val}_I(A)$ or $\mathsf{val}_I(\neg B))$ implies $0)$ iff $((\mathsf{val}_I(B)$ implies $\mathsf{val}_I(C))$ or not $\mathsf{val}_I(A))$
$= ((1$ or not $\mathsf{val}_I(B))$ implies $0)$ iff $((0$ implies $0)$ or not $1)$
$= ((1$ or not $0)$ implies $0)$ iff $(1$ or $0)$
$= ((1$ or $1)$ implies $0)$ iff 1
$= (1$ implies $0)$ iff 1
$= (1$ implies $0)$ iff 1
$= 0$ iff 1
$= 0$

(c) Wir berechnen den Wert der Formel für alle Interpretationen mittels einer Wahrheitstafel. An dieser lassen sich dann die Eigenschaften der Formel ablesen.

A	B	C	$(((A \vee \neg B) \supset C) \equiv ((B \supset C) \vee \neg A))$						
0	0	0	1	1	0	**0**	1	1	1
0	0	1	1	1	1	**1**	1	1	1
0	1	0	0	0	1	**1**	0	1	1
0	1	1	0	0	1	**1**	1	1	1
1	0	0	1	1	0	**0**	1	1	0
1	0	1	1	1	1	**1**	1	1	0
1	1	0	1	0	0	**1**	0	0	0
1	1	1	1	0	1	**1**	1	1	0

Die Formel ist somit erfüllbar und widerlegbar, aber weder gültig noch unerfüllbar.

Aufgabe 6 **(3 Punkte)**

Zeigen Sie, dass die beiden Formeln

$$\neg((P \wedge Q) \supset (\neg P \equiv Q)) \quad \text{und} \quad ((Q \equiv R) \supset P) \wedge \neg(P \uparrow Q)$$

äquivalent sind, und zwar

(a) mit Hilfe einer Wahrheitstafel.

(b) durch algebraische Umformungen.

Lösung

(a) Wahrheitstafel:

P	Q	R	$\neg$	$((P \land Q)$	$\supset$	$(\neg P$	$\equiv Q))$	$=$	$((Q \equiv R)$	$\supset P)$	$\land$	$\neg$	$(P \uparrow Q)$
0	0	0	**0**	0	1	1	0	✓	1	0	**0**	0	1
0	0	1	**0**	0	1	1	0	✓	0	1	**0**	0	1
0	1	0	**0**	0	1	1	1	✓	0	1	**0**	0	1
0	1	1	**0**	0	1	1	1	✓	1	0	**0**	0	1
1	0	0	**0**	0	1	0	1	✓	1	1	**0**	0	1
1	0	1	**0**	0	1	0	1	✓	0	1	**0**	0	1
1	1	0	**1**	1	0	0	0	✓	0	1	**1**	1	0
1	1	1	**1**	1	0	0	0	✓	1	1	**1**	1	0

Da beide Formeln in sämtlichen Wahrheitsbelegungen denselben Wert liefern, sind sie äquivalent.

Tatsächlich ist es gar nicht notwendig, die Wahrheitstabelle vollständig zu befüllen. Wir wissen zum Beispiel, dass $\land$ den Wert 0 liefert, wenn ein Argument den Wert 0 besitzt. Damit ist das Ergebnis der zweiten Formel für jene Interpretationen, in denen $\neg(P \uparrow Q)$ den Wert 0 besitzt, bereits mit 0 festgelegt.

P	Q	R	$\neg$	$((P \land Q)$	$\supset$	$(\neg P$	$\equiv Q))$	$=$	$((Q \equiv R)$	$\supset P)$	$\land$	$\neg$	$(P \uparrow Q)$
0	0	0	**0**	0	1	1	0	✓			**0**	0	1
0	0	1	**0**	0	1	1	0	✓			**0**	0	1
0	1	0	**0**	0	1	1	1	✓			**0**	0	1
0	1	1	**0**	0	1	1	1	✓			**0**	0	1
1	0	0	**0**	0	1	0	1	✓			**0**	0	1
1	0	1	**0**	0	1	0	1	✓			**0**	0	1
1	1	0	**1**	1	0	0	0	✓	0	1	**1**	1	0
1	1	1	**1**	1	0	0	0	✓	1	1	**1**	1	0

(b) Wir vereinfachen beide Formeln. Da wir dabei idente Formeln erhalten, sind die ursprünglichen Formeln äquivalent.

$$
\begin{aligned}
&\neg((P \wedge Q) \supset (\neg P \equiv Q)) && F \equiv G = (F \wedge G) \vee (\neg F \wedge \neg G)\\
&= \neg((P \wedge Q) \supset ((\neg P \wedge Q) \vee (\neg\neg P \wedge \neg Q))) && \neg\neg F = F\\
&= \neg((P \wedge Q) \supset ((\neg P \wedge Q) \vee (P \wedge \neg Q))) && F \supset G = \neg F \vee G\\
&= \neg(\neg(P \wedge Q) \vee (\neg P \wedge Q) \vee (P \wedge \neg Q)) && \neg(F \vee G) = \neg F \wedge \neg G\\
&= \neg\neg(P \wedge Q) \wedge \neg(\neg P \wedge Q) \wedge \neg(P \wedge \neg Q) && \neg\neg F = F\\
&= (P \wedge Q) \wedge \neg(\neg P \wedge Q) \wedge \neg(P \wedge \neg Q) && \neg(F \wedge G) = \neg F \vee \neg G\\
&= (P \wedge Q) \wedge (\neg\neg P \vee \neg Q) \wedge (\neg P \vee \neg\neg Q) && \neg\neg F = F\\
&= P \wedge Q \wedge (P \vee \neg Q) \wedge (\neg P \vee Q) && F \wedge (F \vee G) = F\\
&= P \wedge Q \wedge (\neg P \vee Q) && F \wedge (F \vee G) = F\\
&= P \wedge Q
\end{aligned}
$$

$$
\begin{aligned}
&((Q \equiv R) \supset P) \wedge \neg(P \uparrow Q) && F \uparrow G = \neg F \vee \neg G\\
&= ((Q \equiv R) \supset P) \wedge \neg(\neg P \vee \neg Q) && \neg(F \vee G) = \neg F \wedge \neg G\\
&= ((Q \equiv R) \supset P) \wedge \neg\neg P \wedge \neg\neg Q && \neg\neg F = F\\
&= ((Q \equiv R) \supset P) \wedge P \wedge Q && F \supset G = \neg F \vee G\\
&= (\neg(Q \equiv R) \vee P) \wedge P \wedge Q && (G \vee F) \wedge F = F\\
&= P \wedge Q
\end{aligned}
$$

Im letzten Schritt steht F für die Formel P und G für die Formel $\neg(Q \equiv R)$, wir vereinfachen somit $P \wedge (P \vee \neg(Q \equiv R))$ zu P (modulo Kommutativität).

Aufgabe 7 (3 Punkte)

Gegeben seien folgende zwei Sachverhalte:

(a) Sokrates sagt: „Wenn ich schuldig bin, muss ich bestraft werden. Ich bin schuldig.“
Kann man daraus schließen, dass Sokrates bestraft werden muss?

(b) Sokrates sagt: „Wenn ich schuldig bin, muss ich bestraft werden. Ich muss bestraft werden.“
Kann man daraus schließen, dass Sokrates schuldig ist?

Verwenden Sie für Ihre Überlegungen die Aussagenlogik.
Wie sieht jeweils die Formel aus, deren Gültigkeit/Nichtgültigkeit zeigen würde, dass die Konsequenzbeziehung gilt/nicht gilt?

Lösung

Zur aussagenlogischen Modellierung führen wir Variablen mit folgender Bedeutung ein.
S … Sokrates ist schuldig.
B … Sokrates muss bestraft werden.

(a) Wir übersetzen die Aussagen in logische Formeln:

- $F_1 = S \supset B$ … Wenn Sokrates schuldig ist, muss er bestraft werden.
- $F_2 = S$ … Sokrates ist schuldig.

Daraus wollen wir schließen:

- $F_3 = B \ldots$ Sokrates muss bestraft werden.

Es geht also darum zu überprüfen, ob die Konsequenzbeziehung

$$F_1, F_2 \models F_3$$

gilt.

$I(S)$	$I(B)$	F_1,	F_2	$\models_I$	F_3
0	0	1	0	✓	0
0	1	1	0	✓	1
1	0	0	1	✓	0
1	1	1	1	✓	1

Man kann also aus den gegebenen Argumenten schließen, dass Sokrates bestraft werden muss. Oder anders formuliert: Die Formel B ist eine logische Konsequenz der Prämissen $S \supset B$ und S.

Formel zur Konsequenzbeziehung: F_3 ist genau dann eine logische Konsequenz der Formeln F_1 und F_2, wenn die Formel

$$(F_1 \wedge F_2) \supset F_3$$

gültig ist.

(b) Wir übersetzen die Aussagen in logische Formeln:

- $F_1 = S \supset B \ldots$ Wenn Sokrates schuldig ist, muss er bestraft werden.
- $F_2 = B \ldots$ Sokrates muss bestraft werden.

Daraus wollen wir schließen:

- $F_3 = S \ldots$ Sokrates ist schuldig.

Es geht also darum zu überprüfen, ob die Konsequenzbeziehung

$$F_1, F_2 \models F_3$$

gilt.

$I(S)$	$I(B)$	F_1,	F_2	$\models_I$	F_3
0	0	1	0	✓	0
0	1	1	1	$\lightning$	0
1	0	0	0	✓	1
1	1	1	1	✓	1

Man kann also aus den gegebenen Argumenten nicht schließen, dass Sokrates schuldig ist. Oder anders formuliert: Die Formel S ist keine logische Konsequenz der Prämissen $S \supset B$ und B.

Formel zur Konsequenzbeziehung: F_3 ist genau dann eine logische Konsequenz der Formeln F_1 und F_2, wenn die Formel

$$(F_1 \wedge F_2) \supset F_3$$

gültig ist.

Aufgabe 8 (2 Punkte)

Sei f folgende dreistellige Funktion.

x	y	z	$f(x,y,z)$
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	0

x	y	z	$f(x,y,z)$
0	1	1	0
0	1	0	0
0	0	1	1
0	0	0	0

Stellen Sie f durch eine Formel in

(a) disjunktiver

(b) konjunktiver

Normalform dar.

Lösung

(a) $(A_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge A_2 \wedge \neg A_3) \vee (A_1 \wedge \neg A_2 \wedge A_3) \vee (\neg A_1 \wedge \neg A_2 \wedge A_3)$

(b) $(\neg A_1 \vee A_2 \vee A_3) \wedge (A_1 \vee \neg A_2 \vee \neg A_3) \wedge (A_1 \vee \neg A_2 \vee A_3) \wedge (A_1 \vee A_2 \vee A_3)$

Aufgabe 9 (2 Punkte)

Sei X die Formel $(((G \vee \neg F) \supset H) \vee (\neg F \supset G))$.

(a) Bestimmen Sie eine zu X äquivalente Formel in konjunktiver Normalform. Verwenden Sie die semantische Methode.

(b) Bestimmen Sie eine zu X äquivalente Formel in disjunktiver Normalform. Verwenden Sie die algebraische Methode.

Lösung

(a) KNF mittels semantischer Methode:

F	G	H	$((G \vee \neg F) \supset H) \vee (\neg F \supset G)$					
0	0	0	1	1	0	**0**	1	0
0	0	1	1	1	1	**1**	1	0
0	1	0	1	1	0	**1**	1	1
0	1	1	1	1	1	**1**	1	1
1	0	0	0	0	1	**1**	0	1
1	0	1	0	0	1	**1**	0	1
1	1	0	1	0	0	**1**	0	1
1	1	1	1	0	1	**1**	0	1

Aus dieser Tafel lässt sich folgende KNF ablesen:

$$F \vee G \vee H$$

(b) DNF mittels algebraischer Methode:

$$\begin{array}{ll}
((G \vee \neg F) \supset H) \vee (\neg F \supset G) & \scriptstyle A \supset B = \neg A \vee B \\
= (\neg(G \vee \neg F) \vee H) \vee \neg\neg F \vee G & \scriptstyle \neg(A \vee B) = \neg A \wedge \neg B \\
= (\neg G \wedge \neg\neg F) \vee H \vee \neg\neg F \vee G & \scriptstyle \neg\neg A = A \\
= (\neg G \wedge F) \vee H \vee F \vee G & \scriptstyle A \vee (A \wedge B) = A \\
= F \vee G \vee H &
\end{array}$$

Aufgabe 10 (4 Punkte)

In einem Kriminalfall sind drei Verdächtige festgenommen worden. Sherlock Holmes führt die Untersuchung durch und sagt zu Dr. Watson:

> „Mein lieber Watson, meine intensiven Nachforschungen gestatten mir, folgende Schlüsse zu ziehen: Wenn sich Brown oder Cooper als Täter herausstellen sollte, dann ist Adams unschuldig. Ist aber Adams oder Cooper unschuldig, dann muss Brown ein Täter sein. Ist Cooper schuldig, dann wäre Adams Mittäter.“

Wie lautet die Lösung des Falls? Formalisieren Sie die Hinweise mit Hilfe der Aussagenlogik und werten Sie die Formeln geeignet aus.

Lösung

Wir führen folgende Aussagenvariablen ein:

A ... Adams ist der Täter.
B ... Brown ist der Täter.
C ... Cooper ist der Täter.

Nun formalisieren wir die vorhandenen Informationen:

- Wenn sich Brown oder Cooper als Täter herausstellen sollten, dann ist Adams unschuldig.
  $(B \vee C) \supset \neg A$
- Ist aber Adams oder Cooper unschuldig, dann muss Brown ein Täter sein.
  $(\neg A \vee \neg C) \supset B$
- Ist Cooper schuldig, dann wäre Adams Mittäter.
  $C \supset A$

Wir suchen nun alle Wahrheitsbelegungen für die Variablen A, B und C, sodass diese drei Formeln wahr werden.

A	B	C	$(B \vee C) \supset \neg A$	$(\neg A \vee \neg C) \supset B$	$C \supset A$	
0	0	0	1	0	1	
0	0	1	1	0	0	
0	1	0	1	1	1	✓
0	1	1	1	1	0	
1	0	0	1	0	1	
1	0	1	0	1	1	
1	1	0	0	1	1	
1	1	1	0	1	1	

Bei einer Interpretation sind alle Formeln wahr. Brown ist der Täter.

Aufgabe 11 (4 Punkte)

Professor John Frink organisiert eine internationale Konferenz und sucht zur Unterstützung Student Volunteers. Nach zahlreichen Vorstellungsgesprächen schränkt er die Auswahl auf Bart, Janey, Lisa, Milhouse und Richard ein, wobei allerdings nur Lisa und Richard auch Fremdsprachen beherrschen. Er stellt folgende Überlegungen an:

- Janey möchte ich auf jeden Fall, sie hat bei der letzten Konferenz schon erfolgreich mitgearbeitet.
- Ich kann höchstens drei Volunteers anstellen.
- Ich brauche jedenfalls mindestens einen Volunteer, der Fremdsprachen spricht.
- Richard und Milhouse kennen die Räume, in denen die Konferenz stattfinden soll. Einen der beiden sollte ich auf jeden Fall nehmen, aber beide zu nehmen ist nicht notwendig.
- Richard will nur mitmachen, wenn ich auch Bart anstelle.
- Milhouse und Lisa wollen nur gemeinsam genommen werden, da sie andernfalls zusammen auf Urlaub fahren wollen.

(a) Formalisieren Sie die beschriebene Situation inklusive aller Anhaltspunkte mittels aussagenlogischer Formeln. Geben Sie die Bedeutung der von Ihnen verwendeten Aussagenvariablen an.

(b) Welche Möglichkeiten hat Professor Frink sein Personalproblem zu lösen? Begründen Sie Ihre Antwort(en) mit Hilfe Ihrer aussagenlogischen Modellierung.

Lösung

(a) Aussagenvariablen und ihre Bedeutung:

B ... Prof. Frink stellt Bart an.
J ... Prof. Frink stellt Janey an.
L ... Prof. Frink stellt Lisa an.
M ... Prof. Frink stellt Milhouse an.
R ... Prof. Frink stellt Richard an.

Aussagenlogische Formeln:

$F_0 := J$ Janey auf jeden Fall

$F_1 := \neg(B \wedge L \wedge M) \wedge \neg(B \wedge L \wedge R) \wedge \neg(B \wedge M \wedge R) \wedge \neg(L \wedge M \wedge R)$
höchstens zwei weitere Volunteers

oder

$F_1 := \neg(B \wedge J \wedge L \wedge M) \wedge \neg(B \wedge J \wedge L \wedge R) \wedge \neg(B \wedge J \wedge M \wedge R)$
$\wedge \neg(B \wedge L \wedge M \wedge R) \wedge \neg(J \wedge L \wedge M \wedge R)$
höchstens drei Volunteers (nicht vier oder mehr)

$F_2 := R \vee L$ mind. einer der/die Englisch spricht
$F_3 := R \not\equiv M$ entweder Richard oder Milhouse
$F_4 := R \supset B$ Richard nur dann wenn Bart
$F_5 := M \equiv L$ Milhouse und Lisa nur gemeinsam

(b) Wir suchen alle Wahrheitsbelegungen für die Variablen R, J, M, B und L, sodass die Formeln $F_0, \ldots, F_5$ wahr werden. Wegen Formel F_0 und F_1 genügt es jene Belegungen zu betrachten, in denen höchstens drei Variablen wahr sind, wovon eine J ist.

J	R	M	B	L	F_0	F_1	$R \vee L$	$R \not\equiv M$	$R \supset B$	$M \equiv L$	
1	0	0	0	0	1	1	0				
1	0	0	0	1	1	1	1	0			
1	0	0	1	0	1	1	0				
1	0	0	1	1	1	1	1	0			
1	0	1	0	0	1	1	0				
1	0	1	0	1	1	1	1	1	1	1	✓
1	0	1	1	0	1	1	0				
1	1	0	0	0	1	1	1	1	0		
1	1	0	0	1	1	1	1	1	0		
1	1	0	1	0	1	1	1	1	1	1	✓
1	1	1	0	0	1	1	1	0			

Prof. Frink hat die Wahl zwischen Janey, Milhouse und Lisa, oder Janey, Richard und Bart.

Aufgabe 12 **(4 Punkte)**

Angenommen Sie und Ihre Kollegin modellieren das gleiche aussagenlogische Problem. Sie können sich zwar über die benötigten Aussagenvariablen und ihre Bedeutung einigen, für die Problembeschreibung benötigen Sie aber zwei Formeln F und G, während Ihre Kollegin mit einer einzigen Formel H auskommt, die ganz anders aussieht als Ihre Formeln.

(a) Wie können Sie mit Hilfe eines SAT-Solvers überprüfen, ob die beiden Beschreibungen gleichwertig (semantisch äquivalent) sind? Beschreiben Sie alle erforderlichen Schritte.

(b) Was bedeutet es, wenn der SAT-Solver eine erfüllende Variablenbelegung findet? Was lässt sich über den Wahrheitswert der ursprünglichen Formeln F, G und H in dieser Variablenbelegung sagen?

Lösung

(a) die Problembeschreibungen F, G und H sind äquivalent

$\iff$ die Formeln $F \wedge G$ und H sind äquivalent

$\iff$ $(F \wedge G) \equiv H$ gültig

$\iff$ $(F \wedge G) \not\equiv H$ unerfüllbar

[$\iff$ $((F \wedge G) \vee H) \wedge (\neg(F \wedge G) \vee \neg H)$ unerfüllbar]

[$\iff$ $(F \vee H) \wedge (G \vee H) \wedge (\neg F \vee \neg G \vee \neg H)$ unerfüllbar]

(Für manche SAT-Solver muss die eingegebene Formel zuerst in KNF gebracht werden.)

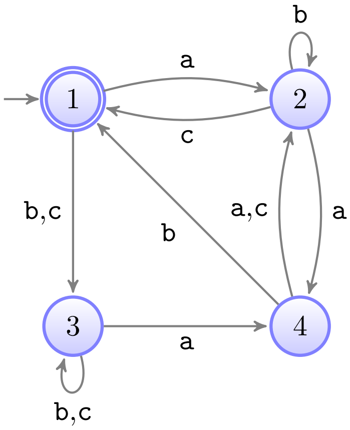
Liefert der SAT-Solver für die letzte Formel die Antwort ‘nein, ist nicht erfüllbar’, dann sind die beiden ursprünglichen Beschreibungen äquivalent. Liefert der SAT-Solver die Antwort ‘ja, ist erfüllbar’, dann sind sie nicht äquivalent.

(b) Wenn der SAT-Solver eine erfüllende Variablenbelegung I findet, dann sind die ursprünglichen Beschreibungen nicht äquivalent. I ist ein Gegenbeispiel für die Äquivalenz, d.h., die Formel $(F \wedge G) \equiv H$ ist falsch in I. Das trifft in folgenden Situationen zu.

$\mathsf{val}_I(F)$	$\mathsf{val}_I(G)$	$\mathsf{val}_I(H)$
0	0	1
0	1	1
1	0	1
1	1	0

Aufgabe 13 (3 Punkte)

Sei $\mathcal{A}$ der folgende endliche Automat.



(a) Geben Sie alle Wörter an, die aus maximal drei Zeichen bestehen und von $\mathcal{A}$ akzeptiert werden. (Das sollten 6 Wörter sein.)

(b) Berechnen Sie schrittweise $\delta^*(1, \texttt{bbacbc})$.

(c) Spezifizieren Sie $\mathcal{A}$ in tabellarischer Form. Handelt es sich bei $\mathcal{A}$ um einen deterministischen oder indeterministischen Automaten?

Lösung

(a) Die von $\mathcal{A}$ akzeptierten Wörter mit einer Länge von maximal drei Zeichen sind ε, `ac`, `aab`, `abc`, `bab`, `cab`.

(b)
$$\begin{aligned}
\delta^*(1, \texttt{bbacbc}) &= \delta^*(\delta(1, \texttt{b}), \texttt{bacbc}) \\
&= \delta^*(3, \texttt{bacbc}) \\
&= \delta^*(\delta(3, \texttt{b}), \texttt{acbc}) \\
&= \delta^*(3, \texttt{acbc}) \\
&= \delta^*(\delta(3, \texttt{a}), \texttt{cbc}) \\
&= \delta^*(4, \texttt{cbc}) \\
&= \delta^*(\delta(4, \texttt{c}), \texttt{bc}) \\
&= \delta^*(2, \texttt{bc}) \\
&= \delta^*(\delta(2, \texttt{b}), \texttt{c}) \\
&= \delta^*(2, \texttt{c}) \\
&= \delta^*(\delta(2, \texttt{c}), \varepsilon) \\
&= \delta^*(1, \varepsilon) \\
&= 1
\end{aligned}$$

(c) $\mathcal{A} = \langle\{1, 2, 3, 4\},\ \{\texttt{a}, \texttt{b}, \texttt{c}\},\ \delta,\ 1,\ \{1\}\rangle$, wobei die Übergangsfunktion δ durch folgende

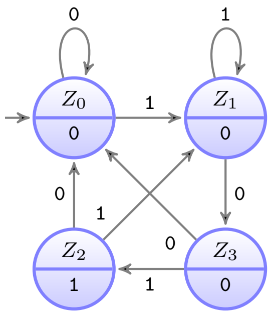
Tabelle definiert ist:

δ	a	b	c
1	2	3	3
2	4	2	1
3	4	3	3
4	2	1	2

$\mathcal{A}$ ist ein deterministischer Automat, da der momentane Zustand und die nächste Eingabe immer eindeutig den Folgezustand bestimmen. Das äußert sich in der Tabelle dadurch, dass jeder Eintrag genau einen Zustand enthält.

Aufgabe 14 (3 Punkte)

Sei $\mathcal{A}$ der folgende Moore-Automat.



(a) Geben Sie die Ausgaben zu folgenden Eingaben an: 11001, 01011, 10101.

(b) Berechnen Sie schrittweise $\delta^*(Z_0, \texttt{01010})$ und $\gamma^*(Z_0, \texttt{01010})$.

(c) Beschreiben Sie die Übersetzungsfunktion $[\mathcal{A}]$.

Lösung

(a)

w:	11001	01011	10101
$[\mathcal{A}](w)$:	000000	000010	000100

(b)
$$\begin{aligned}
\delta^*(Z_0, \texttt{01010}) &= \delta^*(\delta(Z_0, \texttt{0}), \texttt{1010}) &&= \delta^*(Z_0, \texttt{1010}) \\
&= \delta^*(\delta(Z_0, \texttt{1}), \texttt{010}) &&= \delta^*(Z_1, \texttt{010}) \\
&= \delta^*(\delta(Z_1, \texttt{0}), \texttt{10}) &&= \delta^*(Z_3, \texttt{10}) \\
&= \delta^*(\delta(Z_3, \texttt{1}), \texttt{0}) &&= \delta^*(Z_2, \texttt{0}) \\
&= \delta^*(\delta(Z_2, \texttt{0}), \varepsilon) &&= \delta^*(Z_0, \varepsilon) \\
&= Z_0
\end{aligned}$$

$$
\begin{aligned}
\gamma^*(Z_0, \texttt{01010}) &= \gamma(Z_0) \cdot \gamma^*(\delta(Z_0, \texttt{0}), \texttt{1010}) &&= \texttt{0} \cdot \gamma^*(Z_0, \texttt{1010}) \\
&= \texttt{0} \cdot \gamma(Z_0) \cdot \gamma^*(\delta(Z_0, \texttt{1}), \texttt{010}) &&= \texttt{00} \cdot \gamma^*(Z_1, \texttt{010}) \\
&= \texttt{00} \cdot \gamma(Z_1) \cdot \gamma^*(\delta(Z_1, \texttt{0}), \texttt{10}) &&= \texttt{000} \cdot \gamma^*(Z_3, \texttt{10}) \\
&= \texttt{000} \cdot \gamma(Z_3) \cdot \gamma^*(\delta(Z_3, \texttt{1}), \texttt{0}) &&= \texttt{0000} \cdot \gamma^*(Z_2, \texttt{0}) \\
&= \texttt{0000} \cdot \gamma(Z_2) \cdot \gamma^*(\delta(Z_2, \texttt{0}), \varepsilon) &&= \texttt{00001} \cdot \gamma^*(Z_0, \varepsilon) \\
&= \texttt{00001} \cdot \gamma(Z_0) &&= \texttt{000010}
\end{aligned}
$$

(c) Der Automat erkennt die Zeichenfolge `101` in der Eingabe: Wird `101` eingelesen, so ist die Ausgabe `1`, sonst `0`. Dabei wird nach einer `101`-Folge wieder von vorn begonnen: zum Beispiel führt `101101` zur Ausgabe `0001001`, aber `10101` zur Ausgabe `000100`.

Aufgabe 15 (4 Punkte)

Ein elektronisches Tresorschloss besteht aus einem zweistelligen Display sowie den Tasten `+`, `L`, `R`, `Ok` und `Reset`. Jede Stelle des Displays kann eine der drei Ziffern 0, 1 oder 2 anzeigen. Mit jedem Drücken der `+`-Taste ändert sich die Anzeige der aktiven Stelle von 0 auf 1, von 1 auf 2 bzw. von 2 auf 0. Welche der beiden Stellen aktiv ist, lässt sich durch die `L`- und `R`-Taste kontrollieren: Ein- oder mehrmaliges Drücken der `L`- bzw. `R`-Taste aktiviert die linke bzw. rechte Stelle. Im Anfangszustand zeigt das Display die Zahl 00 an und die linke Stelle ist aktiviert. Wird die Zahl 21 eingestellt und anschließend die `Ok`-Taste gedrückt, öffnet das Schloss; bei jeder anderen Zahl geht das Schloss in einen Fehlerzustand. Sowohl im geöffneten Zustand als auch im Fehlerzustand werden alle weiteren Tasten ausgenommen `Reset` ignoriert, d.h., sie beeinflussen den Zustand des Schlosses nicht. Wird zu einem beliebigen Zeitpunkt die `Reset`-Taste gedrückt, geht das Schloss wieder in den Anfangszustand über. Das Schloss lässt sich zum Beispiel mit jeder der beiden folgenden Tastenkombinationen öffnen:

```
+ + R + Ok
+ Reset + L R + L + Ok Ok
```

(a) Überlegen Sie, welche Informationen notwendig sind, um den Zustand des Schlosses zu beschreiben. Wieviele Zustände kann das Schloss annehmen? Wieviele Zustände sind es im Allgemeinen, wenn das Schloss n Ziffern (statt 3) pro Stelle sowie k Stellen (statt 2) besitzt?

(b) Legen Sie die möglichen Aktionen fest, die zu einem Zustandswechsel führen.

(c) Geben Sie einen endlichen Automaten an, der das Verhalten des beschriebenen Schlosses vollständig beschreibt. Die Sprache des Automaten sollen genau jene Tastenkombinationen sein, die den Tresor öffnen.

Es ist vermutlich übersichtlicher, die Übergangsfunktion des Automaten mittels einer Tabelle statt graphisch zu spezifizieren.

Lösung

Der Zustand des Schlosses wird durch die angezeigten Ziffern sowie durch die Position der Aktivierung eindeutig festgelegt. Daher besitzt das Schloss $3^2 \cdot 2 + 2 = 20$ Zustände, im Allgemeinen sind es $n^k \cdot k + 2$ Zustände. (Die beiden Extrazustände sind der Fehlerzustand und das geöffnete Schloss.) Die Aktionen, die zu Zustandswechseln führen (können), sind die möglichen Tastendrücke, also `+`, `L`, `R`, `Ok` und `Reset`. Als Zustandsbezeichnung wählen wir $\underline{a}b$ bzw. $a\underline{b}$, wobei ab den angezeigten Ziffern entspricht und die Unterstreichung die aktivierte Stelle markiert.
Das Verhalten des Schlosses wird durch den Automaten

$$\langle\{\underline{0}0, \dots, 2\underline{2}, \mathit{Fehler}, \mathit{Offen}\}, \{\texttt{+}, \texttt{L}, \texttt{R}, \texttt{Ok}, \texttt{Reset}\}, \delta, \underline{0}0, \{\mathit{Offen}\}\rangle$$

beschrieben, wobei die Übergangsfunktion δ durch folgende Tabelle definiert wird.

δ	`+`	`L`	`R`	`Ok`	`Reset`
$\underline{0}0$	$\underline{1}0$	$\underline{0}0$	$0\underline{0}$	*Fehler*	$\underline{0}0$
$\underline{1}0$	$\underline{2}0$	$\underline{1}0$	$1\underline{0}$	*Fehler*	$\underline{0}0$
$\underline{2}0$	$\underline{0}0$	$\underline{2}0$	$2\underline{0}$	*Fehler*	$\underline{0}0$
$\underline{0}1$	$\underline{1}1$	$\underline{0}1$	$0\underline{1}$	*Fehler*	$\underline{0}0$
$\underline{1}1$	$\underline{2}1$	$\underline{1}1$	$1\underline{1}$	*Fehler*	$\underline{0}0$
$\underline{2}1$	$\underline{0}1$	$\underline{2}1$	$2\underline{1}$	*Offen*	$\underline{0}0$
$\underline{0}2$	$\underline{1}1$	$\underline{0}2$	$0\underline{2}$	*Fehler*	$\underline{0}0$
$\underline{1}2$	$\underline{2}1$	$\underline{1}2$	$1\underline{2}$	*Fehler*	$\underline{0}0$
$\underline{2}2$	$\underline{0}1$	$\underline{2}2$	$2\underline{2}$	*Fehler*	$\underline{0}0$
$0\underline{0}$	$0\underline{1}$	$\underline{0}0$	$0\underline{0}$	*Fehler*	$\underline{0}0$
$1\underline{0}$	$1\underline{1}$	$\underline{1}0$	$1\underline{0}$	*Fehler*	$\underline{0}0$
$2\underline{0}$	$2\underline{1}$	$\underline{2}0$	$2\underline{0}$	*Fehler*	$\underline{0}0$
$0\underline{1}$	$0\underline{2}$	$\underline{0}1$	$0\underline{1}$	*Fehler*	$\underline{0}0$
$1\underline{1}$	$1\underline{2}$	$\underline{1}1$	$1\underline{1}$	*Fehler*	$\underline{0}0$
$2\underline{1}$	$2\underline{2}$	$\underline{2}1$	$2\underline{1}$	*Offen*	$\underline{0}0$
$0\underline{2}$	$0\underline{0}$	$\underline{0}2$	$0\underline{2}$	*Fehler*	$\underline{0}0$
$1\underline{2}$	$1\underline{0}$	$\underline{1}2$	$1\underline{2}$	*Fehler*	$\underline{0}0$
$2\underline{2}$	$2\underline{0}$	$\underline{2}2$	$2\underline{2}$	*Fehler*	$\underline{0}0$
Fehler	*Fehler*	*Fehler*	*Fehler*	*Fehler*	$\underline{0}0$
Offen	*Offen*	*Offen*	*Offen*	*Offen*	$\underline{0}0$