Übungen zu Einführung in die Informatik I

Aufgabe 3 Repräsentation, Interpretation und Information (Lösungsvorschlag)

- a) „Rudolf arbeitet an Helgas Rechner und Helgas Rechner funktioniert nicht mehr“
wird u.U. anders verstanden als
„Helgas Rechner funktioniert nicht mehr und Rudolf arbeitet an Helgas Rechner“

Das umgangssprachliche „und“ enthält oft eine kausale Abhängigkeit im Sinne von „und deshalb“. Die umgangssprachliche „und“-Verknüpfung ist deshalb, im Gegensatz zum logischen *and* im allgemeinen nicht kommutativ.

- b) Das umgangssprachliche „und“ in der Aussage C ist nicht präzise genug. Es kann entweder ausdrücken, daß Hans und Helga gemeinsam spazieren gehen oder, daß sowohl Hans als auch Helga getrennt von einander spazieren gehen. Bei der Formalisierung müssen diese Alternativen entweder berücksichtigt werden. Eine Möglichkeit in unserem Beispiel ist die Hinzunahme folgender Aussage:

D: „Hans und Helga gehen gemeinsam spazieren“

E: „Hans und Helga gehen getrennt spazieren“

Ansatzweise Formalisierung der ursprünglichen Aussage: $C = (D \wedge \neg E) \vee (\neg D \wedge E)$

Anm.: Die Präzisierung umgangssprachlicher Aussagen ist bei dem Übergang von der informellen Beschreibung der Anforderungen an ein Software-System zu der formalisierten Repräsentation eines ausführbaren Programmes häufig erforderlich.

Aufgabe 4 Eigenschaften Boolescher Terme (Lösungsvorschlag)

Ein boolescher Term heißt *elementar*, wenn er keine Identifikatoren enthält. Eine Term heißt *atomar*, wenn er nicht aus weiteren Termen zusammengesetzt ist, sonst heißt er *zusammengesetzt*.

- a) elementar, atomar, nicht zusammengesetzt
- b) nicht elementar, atomar, nicht zusammengesetzt
- c) $x = true$ ist kein Boolescher Term!
Das Symbol „=“ wird bei der induktiven Definition boolescher Terme nicht eingeführt!
- d) nicht elementar, nicht atomar, zusammengesetzt
- e) elementar, nicht atomar, zusammengesetzt

Aufgabe 5 Boolesche Terme (Lösungsvorschlag)

- a) • Wahrheitstabeln für xor, nand und nor:

a	b	a xor b
0	0	0
0	L	L
L	0	L
L	L	0

a	b	a nand b
0	0	L
0	L	L
L	0	L
L	L	0

a	b	a nor b
0	0	L
0	L	0
L	0	0
L	L	0

- Darstellung der Operatoren xor, nand und nor durch die Operatoren $\neg$, $\vee$ und $\wedge$:

$$a \text{ xor } b = (a \wedge \neg b) \vee (\neg a \wedge b)$$

$$a \text{ nand } b = \neg(a \wedge b)$$

$$a \text{ nor } b = \neg(a \vee b)$$

- Es gilt $a \text{ nand } b = \neg(a \wedge b)$ und $\neg(a \wedge a) = \neg a$, also kann die Negation $\neg$ durch nand wie folgt ausgedrückt werden:

$$\neg a = a \text{ nand } a \quad (1)$$

Ferner gilt $a \wedge b = \neg(\neg(a \wedge b))$, also folgt mit Gleichungen (1) und $a \text{ nand } b = \neg(a \wedge b)$:

$$a \wedge b = (a \text{ nand } b) \text{ nand } (a \text{ nand } b) \quad (2)$$

Damit wurde also sowohl die Negation als auch der $\wedge$ -Operator durch den nand-Operator ausgedrückt. Da alle booleschen Operatoren durch die Negation und den $\wedge$ -Operator dargestellt werden können, genügt also alleine der nand-Operator, um alle booleschen Operatoren darzustellen. Dies hat besondere Bedeutung für Schaltfunktionen (siehe TGI), da nur ein Typ von Bausteinen ausreicht, um alle logischen Schaltungen zu realisieren.

- b) Die Äquivalenzen zeigen wir mit den Gesetzen der Booleschen Algebra:

(i)

$$((a \wedge b) \vee c) \wedge (\neg c \vee (a \wedge b)) =$$

$$\text{“Kommutativgesetz”} = ((a \wedge b) \vee c) \wedge ((a \wedge b) \vee \neg c)$$

$$\text{“Distributivgesetz”} = (a \wedge b) \vee (c \wedge \neg c)$$

$$\text{“Neutralitätsgesetz”} = (a \wedge b)$$

(ii)

$$\neg(a \vee b) \vee b =$$

$$\text{“Gesetz von de Morgan”} = (\neg a \wedge \neg b) \vee b$$

$$\text{“Distributivgesetz”} = (\neg a \vee b) \wedge (\neg b \vee b)$$

$$\text{“Neutralitätsgesetz”} = (\neg a \vee b)$$

- (iii) im ersten und im vierten Schritt wird die Äquivalenz $a \Rightarrow b = \neg a \vee b$ verwendet.

$$a \Rightarrow (\neg b \wedge c) = \neg a \vee (\neg b \wedge c)$$

$$\text{“Distributivgesetz”} = (\neg a \vee \neg b) \wedge (\neg a \vee c)$$

$$\text{“Kommutativgesetz”} = (\neg b \vee \neg a) \wedge (c \vee \neg a)$$

$$= (b \Rightarrow \neg a) \wedge (\neg c \Rightarrow \neg a)$$

c) **Behauptung:** $M = \{a, b, \neg(a \wedge b)\}$ leistet das Gewünschte.

Beweis: Der Beweis gliedert sich in zwei Schritte:

1. Schritt: wir zeigen, dass alle zweielementigen Teilmengen von M erfüllbar sind.
2. Schritt: wir zeigen, dass M selbst nicht erfüllbar ist.

zu 1: M besitzt die zweielementigen Teilmengen $T_1 = \{a, b\}$, $T_2 = \{a, \neg(a \wedge b)\}$ und $T_3 = \{b, \neg(a \wedge b)\}$. Für T_1 wählen wir die Belegung $\beta(a) = \beta(b) = L$, dann gilt $I_\beta[a] = I_\beta[b] = L$. Für T_2 wählen wir die Belegung $\beta(a) = L$ und $\beta(b) = 0$, dann gilt $I_\beta[a] = L$ und $I_\beta[\neg(a \wedge b)] = \text{not}(I_\beta[(a \wedge b)]) = \text{not}(\text{and}(I_\beta[a], I_\beta[b])) = \text{not}(\text{and}(L, 0)) = L$.

Für T_3 wählen wir die Belegung $\beta(a) = 0$ und $\beta(b) = L$, dann gilt nach analoger Rechnung $I_\beta[b] = L$ und $I_\beta[\neg(a \wedge b)] = L$.

zu 2: hier genügt es zu zeigen, dass für alle Belegungen β über a und b es ein Element x aus M gibt, so dass $I_\beta[x] = 0$ gilt.

$\beta(a)$	$\beta(b)$	$I_\beta[x]$
0	0	$I_\beta[a] = 0$
0	L	$I_\beta[a] = 0$
L	0	$I_\beta[b] = 0$
L	L	$I_\beta[\neg(a \wedge b)] = 0$

□

Aufgabe 6 Spezifikation eines technischen Systems (Lösungsvorschlag)

Anhand dieses einfachen, allgemein verständlichen Beispiels einer technischen Anwendung soll gezeigt werden, daß verschiedene Spezifikationen zur gleichen informell beschriebenen Aufgabenstellung sinnvoll sein können. Zum zweiten soll eine einfache Art der Modellbildung zur Formalisierung der Aufgabe gezeigt werden. Die Aufgabe hat also durchaus mehrere vernünftige Lösungen, hier wird nur eine solche gegeben.

a) Eine gängige Spezifikation ist:

Der Beifahrerairbag soll ausgelöst werden, wenn das Fahrzeug extrem stark entgegen der Fahrtrichtung beschleunigt wird und der Beifahrergurt geschlossen ist oder der Beifahrersitz belastet ist.

Kurz: S_3 an und (S_4 an oder S_5 an)

b) $S_3 \wedge (S_4 \vee S_5)$

c) Betrachtet man zu der Spezifikation von (a) zunächst nur die Sensoren S_3, S_4 und S_5 , so soll der Airbag in den folgenden Zustandskombinationen ausgelöst werden (dabei steht L für "an", 0 für "aus"): $S_3 S_4 S_5$: L L 0, L 0 L, L L L.

Die Zustände von S_1, S_2 , und S_6 spielen keine Rolle. Steht das Zeichen "-" für einen beliebigen Wert, also 0 oder L, so erhält man die Tabelle

S_1	S_2	S_3	S_4	S_5	S_6
-	-	L	L	0	-
-	-	L	0	L	-
-	-	L	L	L	-

Will man “-” nicht verwenden, so ergibt jede dieser Zeilen 8 Zeilen für die verschiedenen Möglichkeiten der Zustände von S_1 , S_2 und S_6 .

Aufgabe 7 Boolesche Ausdrücke in Gofer (Lösungsvorschlag)

- a) Die Wertetafel der Implikation läßt sich wie folgt implementieren:

```
impl1 :: Bool -> Bool -> Bool

impl1 True True = True
impl1 False False = True
impl1 False True = True
impl1 True False = False
```

- b) Bei Verwendung der booleschen Operationen not, && und || ergibt sich die Implementierung:

```
impl2 :: Bool -> Bool -> Bool

impl2 x y = not x || y
```

- c) • Um das gegebene Gesetz mit Gofer zu zeigen, definieren wir eine dreistellige Funktion

```
f :: Bool -> Bool -> Bool -> Bool

f x y z = (impl2 (x && y) z) == (impl2 (impl2 x y) (impl2 x z))
```

Werten wir nun die Funktion

```
istAequivalent = f True True True && f True True False && f True False True
                && f True False False && f False True True && f False True False
                && f False False True && f False False False
```

aus, so ergibt sich `istAequivalent = True` und die Richtigkeit des Gesetzes ist mit Gofer gezeigt.

- Um die Richtigkeit des Gesetzes

$$(x \wedge y) \Rightarrow z = (x \Rightarrow y) \Rightarrow (x \Rightarrow z)$$

algebraisch zu zeigen, betrachten wir den Term rechts des Gleichheitszeichens und formen diesen unter Verwendung der Gesetze der Booleschen Algebra um (machen Sie sich klar, welche Gesetze jeweils eingesetzt wurden!):

$$\begin{aligned} (x \Rightarrow y) \Rightarrow (x \Rightarrow z) &= \neg(\neg x \vee y) \vee (\neg x \vee z) \\ &= (x \wedge \neg y) \vee (\neg x \vee z) \\ &= ((x \wedge \neg y) \vee \neg x) \vee z \\ &= (x \vee \neg x) \wedge (\neg y \vee \neg x) \vee z \\ &= \neg(x \wedge y) \vee z \\ &= (x \wedge y) \Rightarrow z \end{aligned}$$