Klausur TheGI 3

WS 00/01

Aufgabe 1

(a) Untersuchen Sie mit Hilfe der Wahrheitstafelmethode die folgende Formel auf Erfüllbarkeit und Allgemeingültigkeit: $(p \to \neg q) \to ((q \to p) \to \bot)$! (2 P)

Lösung:

p	q	$(p$	$\to$	$\neg$	$q)$	$\to$	$((q$	$\to$	$p)$	$\to$	$\bot)$
T	T	T	F	F	T	T	T	T	T	F	F
T	F	T	T	T	F	F	F	T	T	F	F
F	T	F	T	F	T	T	T	F	F	T	F
F	F	F	T	T	F	F	F	T	F	F	F

Die Formel ist erfüllbar, aber nicht allgemeingültig.

(b) Ergänzen Sie die folgende Wahrheitstafel! (Die Formel dieser Tafel φ ist mit Hilfe der Junktoren aus $\{\top, \bot, \neg, \wedge, \vee, \to, \leftrightarrow\}$ gebildet, und es ist Symb$(\varphi) = \{p, q\}$. Die Wahrheitstafel darf auf dem Aufgabenblatt ergänzt werden.) (2 P)

p	q	...	...	...	((	...	$\to$	q	)	...	...	)
T	T			T						T		
T	F	F	T	T		T				T	T	
F	T		F	F		T						
F	F			T						T	F	

Lösung:

p	q	$\neg$	p	$\to$	((	$\top$	$\to$	q	)	$\to$	p	)
T	T	F	T	T		T	T	T		T	T	
T	F	F	T	T		T	F	F		T	T	
F	T	T	F	F		T	T	T		F	F	
F	F	T	F	T		T	F	F		T	F	

Aufgabe 2
Gegeben sei die Formel $(p \to \neg q) \to ((q \to p) \to \bot)$ aus Aufgabe 1(a).

(a) Geben Sie eine disjunktive Normalform für diese Formel an! (2 P)

Lösung: z.B. q oder $(p \wedge q) \vee (\neg p \wedge q)$

(b) Geben Sie eine konjunktive Normalform für diese Formel an! (2 P)

Lösung: z.B. q oder $(\neg p \vee q) \wedge (p \vee q)$

Aufgabe 3
Untersuchen Sie die folgenden Formeln mit Hilfe des Resolutionsverfahrens auf Erfüllbarkeit!

(a) $(p \vee q) \wedge (\neg p \vee q) \wedge (p \vee \neg q)$. (2 P)

Lösung: $S = \{\{p, q\}, \{\neg p, q\}, \{p, \neg q\}\}$ ist eine Klauselrepräsentation dieser Formel, und aus S kann wie folgt bewiesen werden:

$$\begin{array}{rcl}
\{p, q\},\ \{\neg p, q\} & \xrightarrow[\text{res}]{} & \{q\} \\
\{p, q\},\ \{p, \neg q\} & \xrightarrow[\text{res}]{} & \{p\} \\
\{\neg p, q\},\ \{p, \neg q\} & \xrightarrow[\text{res}]{} & \{p, \neg p\},\ \{q, \neg q\}
\end{array}$$

Weitere Resolutionen ergeben keine neuen Formeln, deshalb ist die gegebene Formel erfüllbar. $\square$

(b) $(p \vee q \vee \neg r) \wedge (\neg p \vee q) \wedge (r \vee p) \wedge (\neg q \vee \neg r) \wedge (q \vee r) \wedge (\neg q \vee \neg p \vee r)$. (2 P)

Lösung: $S = \{\{p, q, \neg r\}, \{\neg p, q\}, \{r, p\}, \{\neg q, \neg r\}, \{q, r\}, \{\neg q, \neg p, r\}\}$ ist eine Klauselrepräsentation dieser Formel, und aus S kann die leere Klausel wie folgt bewiesen werden:

$$\dfrac{\dfrac{\dfrac{\{p, q, \neg r\} \quad \{\neg p, q\}}{\{q, \neg r\}} \quad \{\neg q, \neg r\}}{\{\neg r\}} \qquad \dfrac{\dfrac{\{\neg q, \neg p, r\} \quad \{\neg p, q\}}{\{\neg p, r\}} \quad \{r, p\}}{\{r\}}}{\varnothing}$$

Folglich ist die gegebene Formel kontradiktorisch, also nicht erfüllbar. $\square$

Aufgabe 4
Beweisen oder widerlegen Sie durch Angabe eines Gegenbeispiels die folgenden Behauptungen über $\varphi, \psi, \chi \in \mathsf{Form}(P)$!

(a) Falls $\Vdash (\varphi \to \psi) \vee (\psi \to \varphi)$, so gilt $\varphi \Vdash \psi$ oder $\psi \Vdash \varphi$. (2 P)

Lösung: Die Aussage gilt nicht.

Gegenbeispiel: sei $\varphi = p$, $\psi = \neg p$.
Für jede Belegung $B : P \to \{\mathrm{T}, \mathrm{F}\}$ gilt $B^*((p \to \neg p) \vee (\neg p \to p)) = \mathrm{T}$, denn wenn $B(p) = \mathrm{T}$ dann $B^*(\neg p \to p) = \mathrm{T}$, andernfalls $B(p) = \mathrm{F}$ und $B^*(p \to \neg p) = \mathrm{T}$. Also ist $(p \to \neg p) \vee (\neg p \to p)$ tautologisch, somit $\Vdash (p \to \neg p) \vee (\neg p \to p)$. Offensichtlich gilt aber weder $p \Vdash \neg p$ noch $\neg p \Vdash p$. $\square$

(b) Es gilt $(\chi \leftrightarrow \psi) \leftrightarrow (\chi \leftrightarrow \varphi) \Vdash \psi \leftrightarrow \varphi$. (2 P)

Lösung: Die Aussage gilt.

Für jede Belegung $B : P \to \{\mathrm{T}, \mathrm{F}\}$ gilt

$$
\begin{array}{ll}
 & B^*((\chi \leftrightarrow \psi) \leftrightarrow (\chi \leftrightarrow \varphi)) = \mathrm{T} \\
\Leftrightarrow & B^*(\chi \leftrightarrow \psi) = B^*(\chi \leftrightarrow \varphi) \\
\Leftrightarrow & (B^*(\chi) = B^*(\phi) \text{ und } B^*(\chi) = B^*(\varphi)) \\
 & \text{oder } (B^*(\chi) \neq B^*(\phi) \text{ und } B^*(\chi) \neq B^*(\varphi)) \\
\Rightarrow & B^*(\psi) = B^*(\varphi) \\
\Leftrightarrow & B^*(\psi \leftrightarrow \varphi) = \mathrm{T}
\end{array}
$$

Also gilt: wenn $B^*((\chi \leftrightarrow \psi) \leftrightarrow (\chi \leftrightarrow \varphi)) = \mathrm{T}$ dann auch $B^*(\psi \leftrightarrow \varphi) = \mathrm{T}$, somit $(\chi \leftrightarrow \psi) \leftrightarrow (\chi \leftrightarrow \varphi) \Vdash \psi \leftrightarrow \varphi$. $\square$

In den Aufgaben 5 und 6 sei die logische Signatur durch die folgenden Angaben bestimmt:

$$
\begin{array}{ll}
\Sigma = & \\
sorts: & data, queue \\
opns: & err :\to data \\
 & dat_1 :\to data \\
 & dat_2 :\to data \\
 & new :\to queue \\
 & enq : data\ queue \to queue \\
 & deq : queue \to queue \\
 & front : queue \to data \\
rels: & Nice : \langle queue \rangle
\end{array}
$$

Die Familie von Variablenmengen sei $X = (X_{data}, X_{queue})$ mit $X_{data} = \{d, d_1, d_2, ...\}$ und $X_{queue} = \{q, q_1, q_2, ...\}$.

Aufgabe 5
Gegeben sei die Substitution $[\sigma]$ mit $\sigma(d_1) = front(q)$, $\sigma(q) = enq(d, q)$. Alle anderen Variablen werden von σ auf sich selbst abgebildet.

(a) Ermitteln Sie, ob $[\sigma]$ für die Formel φ_1:

$$\forall d.(deq(enq(d_1, q)) = enq(d, new)) \vee \exists q.(Nice(q) \leftrightarrow Nice(q_1))$$

zulässig ist und geben Sie im positiven Fall $\varphi_1[\sigma]$ an. Bestimmen Sie im negativen Fall eine geeignete Umbenennung $\langle r\rangle$ und geben Sie $\varphi_1\langle r\rangle[\sigma]$ an! (Eine schrittweise Auswertung von $\varphi_1\langle r\rangle[\sigma]$ ist nicht erforderlich.) (2 P)

Lösung: Die Substitution ist nicht zulässig, denn die Variable q kommt frei im Scope von $\forall d$ vor und $d \in Var(\sigma(q))$.
Eine zulässige Umbenennung ist $\langle r\rangle$ mit $r(d) = d_2$, alle anderen Variablen werden von r auf sich selbst abgebildet.

$$\begin{aligned}
\varphi_2\langle r\rangle[\sigma] &= \forall d_2.(deq(enq(d_1, q)) = enq(d_2, new)) \vee \exists q.(Nice(q) \leftrightarrow Nice(q_1))[\sigma] \\
&= \forall d_2.(deq(enq(front(q), enq(d, q))) = enq(d_2, new)) \\
&\qquad \vee \exists q.(Nice(q) \leftrightarrow Nice(q_1))
\end{aligned}$$

(b) Wie bei (a), nur für die Formel φ_2: (2 P)

$$\forall q.(\forall q_1.(deq(q) = enq(dat_1, q))) \vee \exists q.(Nice(q))$$

Lösung: Die Substitution ist zulässig, da Free$(\varphi_2) = \emptyset$. Dann ist $\varphi_2[\sigma] = \varphi_2$ (nur freie Vorkommen von Variablen werden substituiert).

Aufgabe 6

(a) Gib eine Σ-Struktur A an, in der folgende Formeln gültig sind: (4 P)

(e1) $deq(new) = new$

(e2) $\forall q.((q \neq new) \rightarrow (deq(q) \neq q))$

(e3) $\forall d.(deq(enq(d, new)) = new)$

(e4) $\forall d.(\forall d_1.(\forall q.(deq(enq(d, enq(d_1, q))) = enq(d, deq(enq(d_1, q))))))$

(e5) $front(new) = err$

(e6) $\forall d.(front(enq(d, new)) = d)$

(e7) $\forall d.(\forall d_1.(\forall q.(front(enq(d, enq(d_1, q))) = front(enq(d_1, q)))))$

Lösung:

$$
\begin{array}{ll}
A = & \\
& \begin{array}{ll}
A_{data} = \mathbb{N} & \\
A_{queue} = \mathbb{N}^* & \\
err_A = 0 \in A_{data} & \\
dat_{1,A} = 0 \in A_{data} & \\
dat_{2,A} = 1 \in A_{data} & \\
new_A = \lambda \in A_{queue} & \\
enq_A : & A_{data} \times A_{queue} \to A_{queue} \\
& (d, q) \mapsto dq \\
deq_A : & A_{queue} \to A_{queue} \\
& q \mapsto \begin{cases} \lambda & q = \lambda \\ q_1 & q = q_1 d \end{cases} \\
front_A : & A_{queue} \to A_{data} \\
& q \mapsto \begin{cases} 0 & q = \lambda \\ d & q = q_1 d \end{cases} \\
Nice_A = \mathbb{N}^* \subseteq A_{queue} &
\end{array}
\end{array}
$$

(b) Beweisen Sie, dass die von Ihnen gewählte Struktur A Modell von (e2) ist. Es reicht, die erste und letzte Zeile der rekursiven Auswertung anzugeben. Begründen Sie in einem Satz, warum das Ergebnis ein wahre Aussage darstellt! (2 P)

Lösung: Sei $\beta : X \to A$ beliebig. Dann gilt:

$$
\begin{array}{ll}
& (A, \beta) \models \forall q.((q \neq new) \to (deq(q) \neq q)) \\
\Leftrightarrow & (A, \beta[q|m]) \models q = new \text{ oder } (A, \beta[q|m]) \models deq(q) \neq q \text{ für alle } m \in A_{queue} \\
\Leftrightarrow & m = \lambda \text{ oder } deq_A(m) \neq m \text{ für alle } m \in A_{queue}
\end{array}
$$

Dies gilt, denn entweder $m = \lambda$ oder m hat die Form $m = m_1 ... m_n$ mit $m_i \in \mathbb{N}$, $n \geq 1$, dann ist $deq_A(m) = m_1 ... m_{n-1} \neq m$. □

Aufgabe 7

Gegeben sei eine Signatur Σ mit einer Sorte s, einem Konstantensymbol c und einem einstelligen Relationssymbol P. $X_\Sigma = (X_s)$ sei eine zu Σ passende Familie von Variablenmengen und $x, y \in X_s$. Untersuchen Sie die folgenden Formeln auf Allgemeingültigkeit (Beweis oder Gegenbeispiel, beim Gegenbeispiel reicht die Angabe einer Struktur).

(a) $\exists x. \exists y. (P(x) \leftrightarrow x = y)$. (2 P)

Lösung: Die Formel ist nicht allgemeingültig.

Betrachte eine Struktur A mit $A_s = \{1\}$, $c_A = 1$ und $P_A = \varnothing$. Da $1 \notin P_A$ ist die Formel in A nicht gültig.

(b) $\exists x. \forall y. (y = c \leftrightarrow x = y)$ (2 P)

Lösung: Die Formel ist allgemeingültig.

Sei A eine Σ-Struktur und $\beta : X \to A$ beliebige Variablenbelegung.

$$\begin{array}{ll} & (A,\beta) \models \exists x.\forall y.(y = c \leftrightarrow x = y) \\ \Leftrightarrow & ((a = c_A \text{ und } a = b) \text{ oder } (a \neq c_A \text{ und } a \neq b)) \\ & \quad \text{für ein } b \in A_s \text{ und alle } a \in A_s \end{array}$$

Wähle $b = c_A$, dann gilt für $a = c_A$ auch $a = b$ und für $a \neq c_A$ auch $a \neq b$. Die Formel wird also von β in A bestätigt.

Da dies für jede beliebige Struktur und Variablenbelegung gilt, ist die Formel allgemeingültig. □

Aufgabe 8
Sei Σ eine beliebige Signatur und X eine zu Σ passenden Familie von Variablenmengen.

(a) Zeigen Sie, dass jeder Satz φ die beiden folgenden Eigenschaften erfüllt:

(i) $\mathsf{Mod}_\Sigma(\varphi) \cap \mathsf{Mod}_\Sigma(\neg\varphi) = \varnothing$;

(ii) $\mathsf{Strukt}_\Sigma(X) = \mathsf{Mod}_\Sigma(\varphi) \cup \mathsf{Mod}_\Sigma(\neg\varphi)$. (2 P)

Lösung:

(i) Angenommen es gibt ein $A \in \mathsf{Mod}_\Sigma(\varphi) \cap \mathsf{Mod}_\Sigma(\neg\varphi)$. Dann $A \in \mathsf{Mod}_\Sigma(\varphi)$, also gilt für jede Variablenbelegung $\beta : X \to A : (A,\beta) \models \varphi$. Dann muss aber auch gelten $(A,\beta) \not\models \neg\varphi$, also $A \notin \mathsf{Mod}_\Sigma(\neg\varphi)$. Widerspruch.

Somit gilt $\mathsf{Mod}_\Sigma(\varphi) \cap \mathsf{Mod}_\Sigma(\neg\varphi) = \varnothing$.

(ii) Sei $A \notin \mathsf{Mod}_\Sigma(\varphi)$. Es gibt also eine Variablenbelegungen $\beta : X \to A$ mit $(A,\beta) \not\models \varphi$. Da in φ alle Variablen nur gebunden vorkommen, muss dies für jede Variablenbelegungen β gelten. Dann gilt aber $(A,\beta) \models \neg\varphi$, also $A \in \mathsf{Mod}_\Sigma(\neg\varphi)$. Jede Struktur A ist also Modell für φ oder für $\neg\varphi$, somit gilt $\mathsf{Strukt}_\Sigma(X) = \mathsf{Mod}_\Sigma(\varphi) \cup \mathsf{Mod}_\Sigma(\neg\varphi)$.

(b) Geben Sie ein Beispiel für eine Formel φ an, die (ii) nicht erfüllt (mit Begründung, die Signatur ist frei wählbar). (2 P)

Lösung: Betrachte eine beliebbige Signatur Σ und die Formel $\varphi = (x = y)$. Dann ist $\mathsf{Mod}_\Sigma(\varphi) = \{A \mid |A_{sort(x)}| = 1\}$ und $\mathsf{Mod}_\Sigma(\neg\varphi) = \mathsf{Mod}_\Sigma(x \neq y) = \varnothing$. Also $\mathsf{Mod}_\Sigma(\varphi) \cup \mathsf{Mod}_\Sigma(\neg\varphi) = \mathsf{Mod}_\Sigma(\varphi) \neq \mathsf{Strukt}_\Sigma(X)$. □