Klausur Formale Systeme

Universität Karlsruhe

Fakultät für Informatik

WS 2005/2006

Prof. Dr. P. H. Schmitt

13. April 2006

Name: ______________________

Vorname: ______________________

Matrikel-Nr.: ______________________

Bitte geben Sie auf jedem benutzten Blatt rechts oben Ihren Namen und Ihre Matrikel-Nummer an!

A1 (12)	A2 (7)	A3 (5)	A4 (4)	A5 (4)	A6 (6)	A7 (5)	A8 (8)	A9 (9)	Σ (60)

Bewertungstabelle bitte frei lassen !!!

Zum Bestehen der Klausur benötigen Sie 20 der erreichbaren 60 Punkte.

Gesamtpunkte:

1 Zur Einstimmung (12 Punkte)

Kreuzen Sie in den folgenden Tabellen alles Zutreffende an.

Für jede falsche Antwort wird ein halber Punkt abgezogen!
(Dabei werden jedoch keinesfalls weniger als 0 Punkte für jede der drei Teilaufgaben vergeben.)

Hinweise:

- „PL1“ steht für „Prädikatenlogik erster Ordnung (mit Gleichheit $\doteq$)“; auf diese beziehen sich auch die Begriffe „erfüllbar“, „allgemeingültig“ und „unerfüllbar“.
- „AL“ steht für Aussagenlogik
- In Teilaufgabe a. kann eine Formel mehr als eine der genannten Eigenschaften haben. In Teilaufgabe b. und c. *genau* eine.

a.

	keine Formel der PL1	erfüllbar	allgemeingültig	unerfüllbar
$\forall x(p(f(x))) \wedge \neg p(f(y))$				$\times$
$\forall y(p(x) \vee \neg p(x))$		$\times$	$\times$	
$\forall x(x \doteq c)$		$\times$		

wobei gilt:

- f, c sind Funktionssymbole (mit der richtigen Stelligkeit)
- p ist ein Prädikatssymbol (mit der richtigen Stelligkeit)
- x, y sind Variablen

b.

	Richtig	Falsch
In einem geschlossenen AL-Tableau für F über M kommen nur Teilformeln von F oder von Formeln aus M vor	$\times$	
Das Erfüllbarkeitsproblem für Horn-Klauseln liegt in der Komplexitätsklasse P	$\times$	
Enthält ein Büchi-Automat $\mathcal{B}$ keinen Zyklus, dann gilt stets $L^\omega(\mathcal{B}) = \emptyset$	$\times$	
Sei σ eine kollisionsfreie Substitution und ϕ eine PL1-Formel. Dann gilt: ϕ erfüllbar gdw. $\sigma(\phi)$ erfüllbar		$\times$
Sei ϕ_{sk} die Skolemnormalform einer PL1-Formel ϕ. Dann gilt: $\models \phi_{sk} \leftrightarrow \phi$		$\times$
Eine AL-Formel der Form $F_1 \rightarrow F_2$, wobei F_2 ein Atom enthält, das in F_1 nicht vorkommt, kann keine Tautologie sein		$\times$

c. Sind folgende LTL-Formeln allgemeingültig, d.h. gelten in allen omega-Strukturen?

LTL-Formel	Ja	Nein
$\mathbf{X}\,\Box A \leftrightarrow \Box\mathbf{X}\,A$	$\times$	
$A\ \mathrm{U}\ B \leftrightarrow B \vee \Box(A \wedge \Diamond B)$		$\times$
$\Diamond A \rightarrow A$		$\times$

2 Skolem-Normalform (7 Punkte)

Transformieren Sie folgende PL1-Formel schrittweise in Skolemnormalform:

$$\forall x \exists y \forall z \big(p(x) \to q(f(y), z)\big) \to \forall x \exists u \big(p(x) \vee q(f(u), w)\big)$$

<u>**Lösung:**</u>

1. All-Abschluß

$$\equiv \forall w \Big(\ \forall x \exists y \forall z \big(p(x) \to q(f(y), z)\big) \to \forall x \exists u \big(p(x) \vee q(f(u), w)\big) \ \Big)$$

2. Pränexnormalform

   (a) bereinigen

$$\equiv \forall w \Big(\ \forall x_1 \exists y \forall z \big(p(x_1) \to q(f(y), z)\big) \to \forall x_2 \exists u \big(p(x_2) \vee q(f(u), w)\big) \ \Big)$$

   (b) Quantoren nach außen schieben

$$\begin{aligned} &\equiv \forall w \exists x_1 \forall y \exists z \Big(\ \big(p(x_1) \to q(f(y), z)\big) \to \forall x_2 \exists u \big(p(x_2) \vee q(f(u), w)\big) \ \Big) \\ &\equiv \forall w \exists x_1 \forall y \exists z \forall x_2 \exists u \Big(\ \big(p(x_1) \to q(f(y), z)\big) \to \big(p(x_2) \vee q(f(u), w)\big) \ \Big) \end{aligned}$$

3. Skolemisieren

$$\begin{aligned} &\equiv \forall w \forall y \exists z \forall x_2 \exists u \Big(\ \big(p(g_1(w)) \to q(f(y), z)\big) \to \big(p(x_2) \vee q(f(u), w)\big) \ \Big) \\ &\equiv \forall w \forall y \forall x_2 \exists u \Big(\ \big(p(g_1(w)) \to q(f(y), g_2(w, y))\big) \to \big(p(x_2) \vee q(f(u), w)\big) \ \Big) \\ &\equiv \forall w \forall y \forall x_2 \Big(\ \big(p(g_1(w)) \to q(f(y), g_2(w, y))\big) \to \big(p(x_2) \vee q(f(g_3(w, y, x_2)), w)\big) \ \Big) \end{aligned}$$

4. Matrix in KNF transformieren

$$\begin{aligned} &\equiv \forall w \forall y \forall x_2 \Big(\ \neg\big(p(g_1(w)) \to q(f(y), g_2(w, y))\big) \vee \big(p(x_2) \vee q(f(g_3(w, y, x_2)), w)\big) \ \Big) \\ &\equiv \forall w \forall y \forall x_2 \Big(\ \neg\big(\neg p(g_1(w)) \vee q(f(y), g_2(w, y))\big) \vee \big(p(x_2) \vee q(f(g_3(w, y, x_2)), w)\big) \ \Big) \\ &\equiv \forall w \forall y \forall x_2 \Big(\ \big(p(g_1(w)) \wedge \neg q(f(y), g_2(w, y))\big) \vee \big(p(x_2) \vee q(f(g_3(w, y, x_2)), w)\big) \ \Big) \\ &\equiv \forall w \forall y \forall x_2 \Big(\ \big(p(g_1(w)) \vee p(x_2) \vee q(f(g_3(w, y, x_2)), w)\big) \wedge \\ &\qquad \big(\neg q(f(y), g_2(w, y)) \vee p(x_2) \vee q(f(g_3(w, y, x_2)), w)\big) \ \Big) \end{aligned}$$

3 Resolution (5 Punkte)

Beweisen Sie mithilfe des Resolutionskalküls die Unerfüllbarkeit der folgenden Formel:

$$\forall x(p(x, f(x))) \land \forall x \forall y \forall z(\neg p(x, y) \lor \neg p(y, z) \lor p(x, z)) \land \exists x(\neg p(x, f(f(x))))$$

Lösung:

1. Klauselnormalform herstellen

   $\{\ p(x_1, f(x_1))\ \}\quad \{\ \neg p(x_2, y_2),\ \neg p(y_2, z_2),\ p(x_2, z_2)\ \}\quad \{\ \neg p(c, f(f(c)))\ \}$

2. Resolution

1	$\{\ p(x_1, f(x_1))\ \}$	
2	$\{\ \neg p(x_2, y_2),\ \neg p(y_2, z_2),\ p(x_2, z_2)\ \}$	
3	$\{\ \neg p(c, f(f(c)))\ \}$	
4 (1,2)	$\{\ \neg p(f(x_1), z_2),\ p(x_1, z_2)\ \}$	$\sigma = \{x_2/x_1,\ y_2/f(x_1)\}$
5 (4)	$\{\ \neg p(f(x_3), z_2),\ p(x_3, z_2)\ \}$	Variantenbildung
6 (1,5)	$\{\ p(x_3, f(f(x_3)))\ \}$	$\sigma = \{x_1/f(x_3),\ z_2/f(f(x_3))\}$
7 (3,6)	$\Box$	$\sigma = \{x_3/c\}$

4 Unifikation ((1+1)+2 Punkte)

Bei beiden Teilaufgaben a. und b. gilt:
c, d, f, g, h, k, m sind Funktionszeichen und u, v, w, x, y, z Variablen.

a. Geben Sie für die folgenden Paare von Termen einen allgemeinsten Unifikator μ an (als *eine* Substitution und nicht als Verkettung von Substitutionen). Falls es keinen allgemeinsten Unifikator gibt, begründen Sie!

i.

$m(f(g(x), z), y)$
$m(f(y, h(y, c)), x)$

Lösung:

Nicht unifizierbar. Nach 3 Schritten des Robinson-Algorithmus erhält man

$$\{m(f(g(x), h(g(x), c)), g(x)), m(f(g(x), h(g(x), c)), x)\} \quad ,$$

was sich nicht unifizieren läßt, weil $x \in \mathit{Var}(g(x))$ (occur check).

ii.

$h(f(x, y), f(z, u))$
$h(f(c, g(x, z)), f(c, w))$

Lösung:

$$\mu = \{\ x/c,\ y/g(c, c),\ z/c,\ u/w\ \}$$

b. Zeigen Sie, daß die Substitution

$$\sigma = \{\ u/f(c),\ w/f(c),\ v/g(d),\ y/d,\ z/d\ \}$$

kein allgemeinster Unifikator der folgenden Terme ist:

$k(f(y), w, g(z))$
$k(u, u, v)$

Lösung:

σ ist überhaupt kein Unifikator:

$$\sigma(k(f(y), w, g(z))) = k(f(d), f(c), g(d)) \neq k(f(c), f(c), g(d)) = \sigma(k(u, u, v))$$

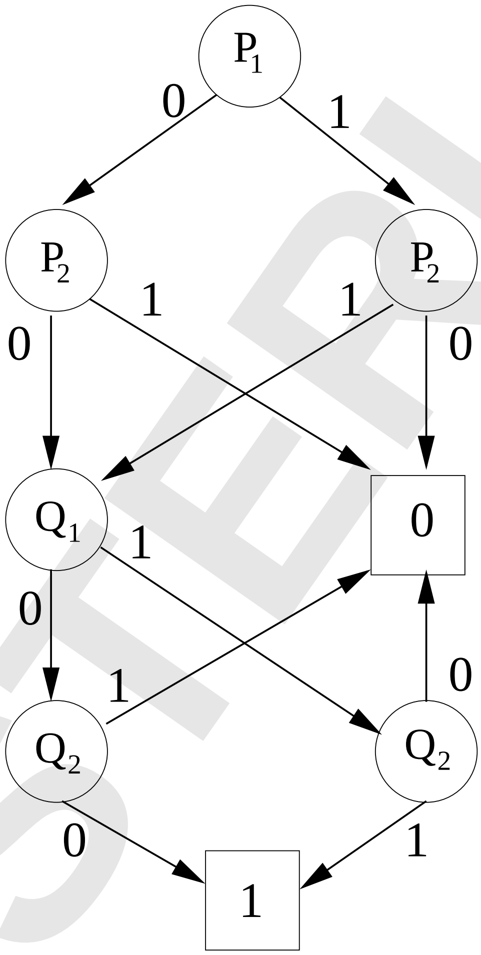
5 Shannongraphen (4 Punkte)

Geben Sie einen *reduzierten* Shannongraphen für die aussagenlogische Formel

$$(P_1 \leftrightarrow P_2) \wedge (Q_1 \leftrightarrow Q_2)$$

an. Verwenden Sie dabei die Variablenordnung $P_1 < P_2 < Q_1 < Q_2$.

Lösung:



6 Semantik der Prädikatenlogik (2+1+3 Punkte)

Definition 1 (Nullsemantik) *Sei M eine Menge prädikatenlogischer Formeln und F eine prädikatenlogische Formel. Wir sagen F ist eine logische Folgerung aus M in der Nullsemantik, genau dann wenn jedes Modell von M, wobei das Universum von M auch die leere Menge sein darf, auch ein Modell von F ist.*

Für die folgenden Teilaufgaben a.–c. sei Σ eine PL1-Signatur, die keine 0-stelligen Funktionssymbole (d.h. Konstanten) enthält.

a. Geben Sie ein Beispiel für eine Formel über Σ an, die in der üblichen Semantik allgemeingültig ist, aber in der Nullsemantik nicht.

Lösung:

$(\forall x p(x)) \rightarrow \exists x p(x)$

b. Wenn man den in der Vorlesung beschrieben Tableaukalkül unverändert für die Nullsemantik benutzt, welche Eigenschaft geht dabei verloren: die Vollständigkeit oder die Korrektheit?

Lösung:

Korrektheit geht verloren.

c. Wie muß das Tableauverfahren abgeändert werden, damit es vollständig und korrekt bezüglich der Nullsemantik ist? Begründen Sie Ihre Antwort.

Lösung:

Ein korrekter und vollständiger Kalkül für die Nullsemantik entsteht, wenn die Pfadabschlußregel abgeändert wird zu:
Eine Substitution σ *schließt* einen Pfad π im Tableau T, wenn es

- Formeln B, C gibt, so daß $\sigma(B) = \sigma(C)$, σ kollisionsfrei für B und C ist und $1B, 0C$ auf π liegen und **zusätzlich** $\sigma(B)$ eine Grundinstanz ist oder
- Terme s, t gibt, so daß $\sigma(s) = \sigma(t)$ und $0s \doteq t$ auf π liegt und **zusätzlich** $\sigma(s)$ ein Grundterm ist.

7 Formalisieren in LTL (1+1+1+2 Punkte)

In dieser Aufgabe soll die Schaltung einer Verkehrsampel formalisiert werden. Die Ampel hat drei Lichter *red*, *yellow*, und *green*.
Zur Formalisierung dürfen Sie ausschließlich die aussagenlogischen Atome R, Y, G mit folgender Bedeutung verwenden:

- R für Lampe *red* ist an
- Y für Lampe *yellow* ist an
- G für Lampe *green* ist an

Im Normalbetrieb befindet sich die Ampel zu jedem Zeitpunkt in einem der folgenden Zustände:

- Zustand 1: *red* an, *yellow* aus, *green* aus
- Zustand 2: *red* an, *yellow* an, *green* aus
- Zustand 3: *red* aus, *yellow* aus, *green* an
- Zustand 4: *red* aus, *yellow* an, *green* aus

a. Drücken Sie in LTL aus, daß die Ampel zu jedem Zeitpunkt in genau einem der Zustände 1–4 ist.

Lösung:

$\Box\big(((R \vee Y) \wedge \neg G) \vee G\big)$

b. Formalisieren Sie in LTL: Am Anfang ist die Ampel im Zustand 1 und nach einer gewissen Zeit wechselt sie in Zustand 2.

Lösung:

$R \wedge \neg Y \wedge \neg G \wedge \big((R \wedge \neg Y \wedge \neg G)\ \mathrm{U}\ (R \wedge Y \wedge \neg G)\big)$

c. Drückt die LTL-Formel

$$\neg\Box R \wedge \neg\Box Y \wedge \neg\Box G$$

aus, daß keine der drei Lampen ununterbrochen leuchtet? Begründen Sie Ihre Antwort!

Lösung:

Ja. Die Formel ist äquivalent zu $\Diamond\neg R \wedge \Diamond\neg Y \wedge \Diamond\neg G$, d.h. es müssen Zeitpunkte existieren, zu denen R, Y, G nicht leuchten.

d. Im Nachtbetrieb gibt es nur einen Zustand, in dem das Licht *yellow* gleichmäßig blinkt. Formalisieren Sie in LTL, daß *yellow* abwechselnd eine Zeiteinheit an und eine Zeiteinheit aus ist.

Lösung:

$\Box(Y \wedge \mathbf{X}\,\neg Y \vee \neg Y \wedge \mathbf{X}\,Y)$

8 LTL und Büchi-Automaten (4+4 Punkte)

Wir erweitern die Syntax der LTL um den zweistelligen Operator *before*.
Die Semantik der LTL-Formel A *before* B ist wie folgt definiert:

Definition 2 *Sei* $\mathcal{R} = (\mathbb{N}, <, \xi)$ *eine omega-Struktur und* A, B *LTL-Formeln.*

$\xi \models A$ *before* B *gdw* *für alle* $n \in \mathbb{N}$ *gilt*
falls $\xi_n \models B$ *dann gibt es ein* $m \in \mathbb{N}$ *mit*
$0 \leq m < n$ *und* $\xi_m \models A$

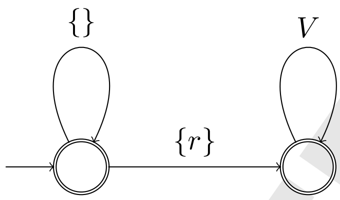
Gegeben sei eine Signatur $\Sigma = \{r, s\}$ und das Automatenalphabet

$$V = 2^{\Sigma} = \{\{\}, \{r\}, \{s\}, \{r, s\}\} \ .$$

a. Geben Sie einen Büchi-Automaten $\mathcal{B}$ an, so daß gilt:

$$L^{\omega}(\mathcal{B}) = \{\xi \in V^{\omega} \mid \xi \models r \textit{ before } s\}$$

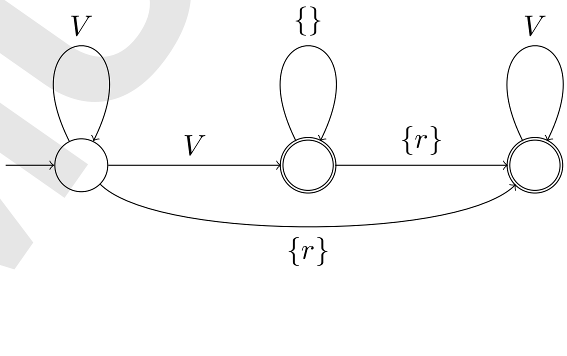
Lösung:



b. Geben Sie einen Büchi-Automaten $\mathcal{B}'$ an, so daß gilt:

$$L^{\omega}(\mathcal{B}') = \{\xi \in V^{\omega} \mid \xi \models \Diamond(r \textit{ before } s)\}$$

Lösung:



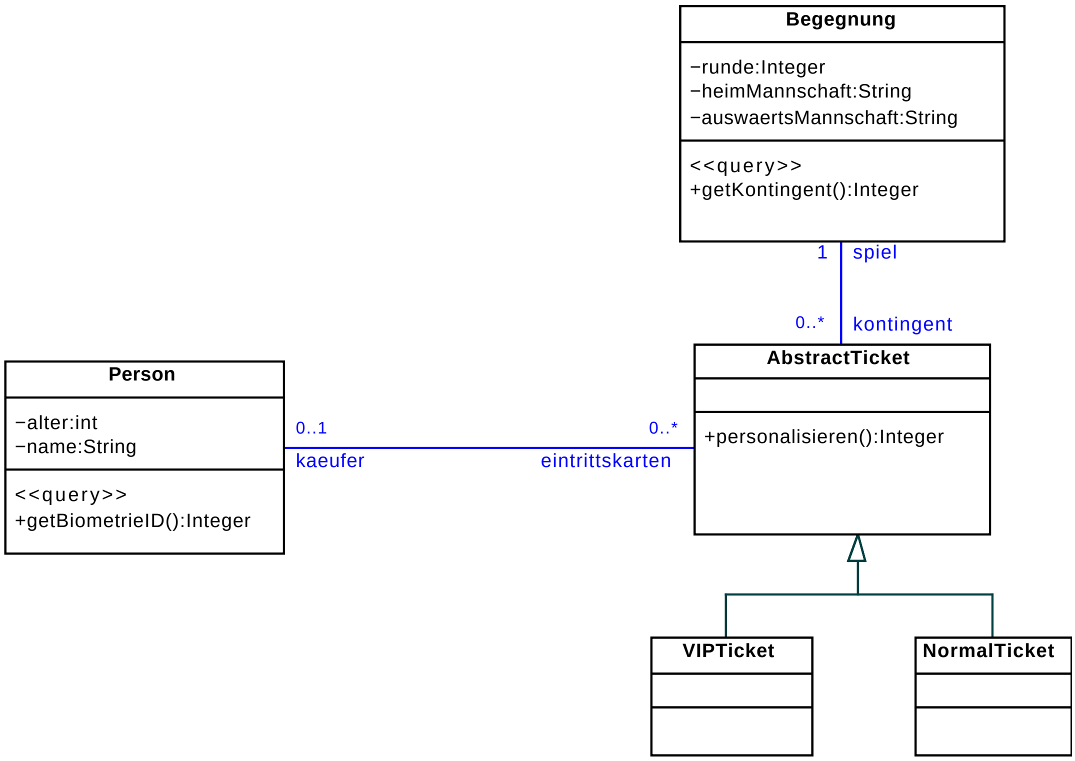


Abbildung zu Aufgabe 9

9 Object Constraint Language ((2 + 2) + (1 + 1) + 3 Punkte)

Gegeben sei das links (auf der Rückseite von Blatt 9) dargestellte UML-Klassendiagramm, welches ein Ticketsystem für die Eintrittskarten der Fußballweltmeisterschaft 2006 modelliert. Die Query `getBiometrieID()` liefert eine Biometrie-Identifikationsnummer als Integerwert zurück, der biometrische Merkmale einer Person kodiert.

Abkürzende Schreibweisen in den OCL-Constraints sind *nicht* erlaubt.

Hinweis: Sie dürfen in dieser Aufgabe die zusätzliche OCL-Operation *expr*.`oclIsKindOf`(*Type*) verwenden, die genau dann zu wahr auswertet, wenn der OCL Ausdruck *expr* ein Objekt vom Typ *Type* beschreibt.

a. Formalisieren Sie die folgenden Anforderungen in OCL:

i. Begegnungen zweier Mannschaften sind pro Runde eindeutig.

Lösung:

```
context Begegnung inv:
Begegnung.allInstances->forAll(b1, b2 |
(b1.heimMannschaft = b2.heimMannschaft and b1.runde = b2.runde
and b1.auswaertsMannschaft = b2.auswaertsMannschaft)
implies b1 = b2)
```

ii. Die Anzahl der VIP Tickets zu einer Begegnung ist auf höchstens 10 Prozent des Gesamtkontingents der Begegnung begrenzt.

Lösung:

```
context Begegnung inv:
self.kontingent->select(t:Ticket |
t.oclIsKindOf(VIPTicket))->size * 10<=self.kontingent->size
```

b. Spezifizieren Sie in OCL das Query `getKontingent()`, welches die Größe des Gesamtkontingents an Karten einer Begegnung liefert, als

i. Methodenspezifikation

Lösung:

```
context Begegnung::getKontingent():Integer
pre : true
post: result = self.kontingent->size
```

ii. Invariante von `Begegnung`

Lösung:

```
context Begegnung inv:
self.getKontingent() = self.kontingent->size
```

c. Für ein verkauftes Ticket berechnet die Methode `personalisieren()` eine Zahl, die dem Produkt aus der Biometrie-Identifikationsnummer des Käufers und der Anzahl der bis vor Aufruf der Methode verkauften Tickets der zugehörigen Begegnung entspricht.

Lösung:

```
context AbstractTicket::personalisieren():Integer
pre : not (self.kaeufer->isEmpty)
post: result = self.kaeufer@pre.getBiometrieID() *
     self.spiel@pre.kontingent@pre->select(t:AbstractTicket|
       not (t.kaeufer@pre)->isEmpty)->size
```