Humboldt-Universität zu Berlin
Lehrstuhl Logik in der Informatik
Prof. Dr. Nicole Schweikardt

28. Januar 2016

Logik in der Informatik

Wintersemester 2015/2016

Übungsblatt 13

Abgabe: *bis 4. Februar, 13.15 Uhr (vor der Vorlesung oder im Briefkasten zwischen den Räumen 3.401 und 3.402 im Johann von Neumann-Haus (Rudower Chaussee 25))*

Aufgabe 1: **(24 Punkte)**

Sei $\sigma := \{R, f_0, f_1, c\}$, wobei c ein Konstantensymbol, R ein 2-stelliges Relationssymbol und f_0, f_1 zwei 1-stellige Funktionssymbole sind.

Beweisen Sie folgende Aussagen aus Korollar 4.41:

(a) Das Unerfüllbarkeitsproblem für $\mathsf{FO}[\sigma]$ ist nicht entscheidbar.

(b) Das Folgerungsproblem für $\mathsf{FO}[\sigma]$ ist nicht entscheidbar.

(c) Das Erfüllbarkeitsproblem für $\mathsf{FO}[\sigma]$ ist nicht semi-entscheidbar.

Aufgabe 2: **(24 Punkte)**

(a) Sei R ein 2-stelliges Relationssymbol, f ein 1-stelliges Funktionssymbol und seien c und d Konstantensymbole.

Im Folgenden ist für jedes $i \in \{1, 2\}$ eine Signatur σ_i und ein $\mathsf{FO}[\sigma_i]$-Satz φ_i gegeben.

(1) Sei $\sigma_1 := \{R, f, c\}$ und sei φ_1 der folgende $\mathsf{FO}[\sigma_1]$-Satz:

$$\forall x \forall y \forall z \left(R(x, f(x)) \ \wedge \ \Big(\big(R(x,y) \wedge R(y,z) \big) \ \rightarrow \ R(x,z) \Big) \right)$$

(2) Sei $\sigma_2 := \{R, c, d\}$ und sei φ_2 der folgende $\mathsf{FO}[\sigma_2]$-Satz:

$$\exists x\, R(c, x) \ \wedge \ \forall x \forall y \big(\neg x{=}y \ \rightarrow R(x,y) \big)$$

Geben Sie für jedes $i \in \{1, 2\}$ eine σ_i-Herbrandstruktur $\mathcal{A}_i$ und eine σ_i-Herbrandstruktur $\mathcal{B}_i$ an, so dass gilt:

$$\mathcal{A}_i \models \varphi_i \qquad \text{und} \qquad \mathcal{B}_i \not\models \varphi_i.$$

Begründen Sie jeweils, warum $\mathcal{A}_i \models \varphi_i$ bzw. $\mathcal{B}_i \not\models \varphi_i$ gilt.

(b) Sei $\sigma := \{f, c\}$, wobei f ein 1-stelliges Funktionssymbol ist und c ein Konstantensymbol.

Zeigen Sie, dass Satz 4.46 aus dem Vorlesungsskript im Allgemeinen *nicht* für $\mathsf{FO}[\sigma]$-Sätze in Skolemform gilt, die *nicht* gleichheitsfrei sind.

Geben Sie dazu einen $\mathsf{FO}[\sigma]$-Satz φ in Skolemform an, so dass gilt:

$$\varphi \text{ ist erfüllbar,} \qquad \text{aber} \qquad \varphi \text{ besitzt kein Herbrand-Modell.}$$

Aufgabe 3: **(27 Punkte)**

Sei $\sigma := \{R, f\}$, wobei R ein 2-stelliges Relationssymbol und f ein 1-stelliges Funktionssymbol ist. Transformieren Sie die $\mathsf{FO}[\sigma]$-Formel

$$\varphi(z) \ := \ \exists x \left(\forall y \ R(x,y) \ \lor \ f(x) = z \right)$$

in einen zu φ erfüllbarkeitsäquivalenten gleichheitsfreien $\mathsf{FO}[\hat{\sigma}]$-Satz $\hat{\varphi}$ in Skolemform.

Hinweis: Gehen Sie dabei vor wie im Beweis von Satz 4.52 im Vorlesungsskript. Geben Sie insbesondere auch die Formel an, die nach jedem der Schritte 1, 2 und 3 des Beweises entsteht.

Weitere Aufgaben finden Sie auf der Rückseite

Aufgabe 4: **(25 Punkte)**

Achtung: *Geben Sie Ihre Lösungsansätze in einer Datei mit dem Namen* `blatt13.pl` *über das GOYA-System ab!* ***Es gilt:*** *Lösungsansätze, die von SWI-Prolog nicht geladen werden können, werden nicht bewertet!*

Wir betrachten im Folgenden gerichtete Graphen. Wir repräsentieren die *Knoten* eines Graphen durch Prolog-Konstanten (d.h., Zahlen oder Atome). Eine *Kante* in einem Graphen repräsentieren wir durch einen Term `(a, b)` für zwei Knoten `a` und `b` des Graphen. Einen *Graphen* repräsentieren wir durch einen Prolog-Term `graph(V, E)`, wobei `V` eine Liste der Knote des Graphen ist, und `E` eine Liste seiner Kanten. Beispiele für solche Repräsentationen von Graphen als Prolog-Terme finden Sie in der Datei `http://www2.informatik.hu-berlin.de/logik/lehre/WS15-16/Logik/downloads/ef/beispiele.pl` .

(a) Implementieren Sie ein Prädikat `iff/2`, so dass für zwei Prolog-Terme `P` und `Q` die Anfrage `?- iff(P, Q).` genau dann erfüllt ist, wenn entweder beide der Anfragen `?- P.` und `?- Q.` erfüllt sind oder keine von beiden.

Hinweis: Nutzen Sie die *Negation as failure* durch den Operator `\+` .

(b) Implementieren Sie ein Prädikat `gb/2`, das die Gewinnbedingungen für Duplicator im EF-Spiel (siehe Seite 153 des Skripts) überprüft.

D.h., für zwei Graphen $\mathcal{A} = (A, E^{\mathcal{A}})$ und $\mathcal{B} = (B, E^{\mathcal{B}})$ sowie zwei Listen `[`a_1`, ..., `a_k`]` und `[`b_1`, ..., `b_k`]` gleicher Länge von Knoten aus $\mathcal{A}$ beziehungsweise $\mathcal{B}$ soll die Anfrage

`?- gb((`$\mathcal{A}$`, [`a_1`, ..., `a_k`]), (`$\mathcal{B}$`, [`b_1`, ..., `b_k`])).`

genau dann erfüllt sein, wenn

(1) für alle $i, j \in \{1, \ldots, k\}$ gilt: $a_i = a_j \iff b_i = b_j$, und

(2) die Abbildung $\pi\colon \{a_1, \ldots, a_k\} \to \{b_1, \ldots, b_k\}$ mit $\pi(a_i) := b_i$ für jedes $i \in \{1, \ldots, k\}$ ein *partieller Isomorphismus* von $\mathcal{A}$ nach $\mathcal{B}$ (gemäß Definition 3.48 des Skripts) ist.

Hinweise: Machen Sie sich mit dem Modul `tupel` vertraut, welches Sie unter der URL `http://www2.informatik.hu-berlin.de/logik/lehre/WS15-16/Logik/downloads/ef/tupel.pl` finden. Nutzen Sie das Modul zur Lösung dieser Teilaufgabe.

Nutzen Sie ggf. das Prädikat `iff/2` aus Teilaufgabe (a) und das Prädikat `forall/2` (siehe `http://www.swi-prolog.org/pldoc/man?predicate=forall/2`).

Bemerkung: Das Prädikat `forall/2` ist in SWI-Prolog vordefiniert, kann aber auch leicht durch die folgende Regel implementiert werden:

```
forall(P, Q) :- \+ (P, \+ Q).
```

(c) Implementieren Sie ein Prädikat `gs/3`, das für ein $m \geq 0$, zwei Graphen $\mathcal{A}$ und $\mathcal{B}$, sowie zwei Tupel $\overline{a}$ und $\overline{b}$ gleicher Länge von Knoten aus $\mathcal{A}$ beziehungsweise $\mathcal{B}$ entscheidet, ob Duplicator eine Gewinnstrategie im m-Runden EF-Spiel auf $(\mathcal{A}, \overline{a})$ und $(\mathcal{B}, \overline{b})$ hat.

D.h., für eine natürliche Zahl `M`, für zwei Graphen $\mathcal{A}$ und $\mathcal{B}$ sowie zwei Listen `[`a_1`, ..., `a_k`]` und `[`b_1`, ..., `b_k`]` gleicher Länge von Knoten aus $\mathcal{A}$ beziehungsweise $\mathcal{B}$ soll die Anfrage

`?- gs(M, (`$\mathcal{A}$`, [`a_1`, ..., `a_k`]), (`$\mathcal{B}$`, [`b_1`, ..., `b_k`])).`

genau dann erfüllt sein, wenn Duplicator eine Gewinnstrategie im m-Runden EF-Spiel auf $(\mathcal{A}, a_1, \ldots, a_k)$ und $(\mathcal{B}, b_1, \ldots, b_k)$ hat.

Hinweis: Führen Sie gegebenenfalls Hilfsprädikate ein. Nutzen Sie Ihre Lösung für die Teilaufgabe (b). Überprüfen Sie die Korrektheit Ihrer Implementation anhand der Beispiele in der Datei `beispiele.pl`.