

Logik in der Informatik

Wintersemester 2015/2016

Übungsblatt 8

Abgabe: bis 17. Dezember, 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)

Betrachten Sie die Kinodatenbank \mathcal{D} aus der Vorlesung.

(a) Geben Sie für die folgenden Anfragen jeweils eine $\text{FO}[\sigma_{\text{KINO}}]$ -Formel φ und ein Variablentupel (x_1, \dots, x_n) mit $\text{frei}(\varphi) \subseteq \{x_1, \dots, x_n\}$ an, die die Anfrage beschreiben. Berechnen Sie jeweils auch die Relation $\llbracket \varphi(x_1, \dots, x_n) \rrbracket^{\mathcal{D}}$.

(i) Geben Sie alle Kombinationen aus Name eines Kinos und Uhrzeit aus, die einer Vorstellung des Films Gravity entsprechen.

(ii) Geben Sie alle Paare von Namen von Kinos aus, die sich im gleichen Stadtteil befinden.

(iii) Geben Sie die Namen aller Kinos aus, in denen mindestens ein Film gezeigt wird, in dem exakt zwei verschiedene Regisseure Regie geführt haben.

(b) Berechnen Sie die Relationen $\llbracket \varphi_1(x) \rrbracket^{\mathcal{D}}$, $\llbracket \varphi_2(x_1, x_2) \rrbracket^{\mathcal{D}}$, $\llbracket \varphi_3(x_1, x_2, x_3) \rrbracket^{\mathcal{D}}$ und geben Sie umgangssprachlich an, welche Anfragen durch die Formeln φ_1 , φ_2 und φ_3 beschrieben werden.

(i) $\varphi_1 := \exists x_F R_{\text{Prog}}(\text{'Movimiento'}, x_F, x)$

(ii) $\varphi_2 := \exists x_{S_1} \exists x_T (R_{\text{Kino}}(x_1, x_2, x_{S_1}, x_T) \wedge \exists x_R \exists x_{S_2} R_{\text{Film}}(x_1, x_R, x_{S_2}))$

(iii) $\varphi_3 := \left(\exists x_K R_{\text{Prog}}(x_K, x_2, x_3) \wedge \exists x_{S_1} (R_{\text{Film}}(x_2, x_1, x_{S_1}) \wedge \forall x_F \forall x_{S_2} (R_{\text{Film}}(x_F, x_1, x_{S_2}) \rightarrow x_F = x_2)) \right)$

Aufgabe 2:**(26 Punkte)**

- (a) Sei die Signatur $\sigma := \{E, f\}$. Hierbei ist E ein zweistelliges Relationssymbol und f ein einstelliges Funktionssymbol. Welche der folgenden Aussagen sind korrekt, welche nicht? (Sie brauchen Ihre Antwort nicht zu begründen.)

(i) $\forall x \exists y f(x)=y \equiv \exists y \forall x f(x)=y$

(ii) $\forall x \forall y (E(x, y) \rightarrow E(y, x)) \equiv \forall y \forall x (\neg E(y, x) \rightarrow \neg E(x, y))$

(iii) $\exists x \exists y f(x)=y \equiv \forall x \exists y ((x=y \vee E(x, y)) \rightarrow \exists z (z=y \vee E(z, y)))$

- (b) Welche der folgenden Aussagen sind für alle Signaturen σ und alle FO[σ]-Formeln φ und ψ korrekt, welche nicht? (Sie brauchen Ihre Antwort nicht zu begründen.)

(i) $(\forall x \varphi \wedge \forall x \psi) \equiv \forall x (\varphi \wedge \psi)$

(iii) $(\exists x \varphi \wedge \exists x \psi) \equiv \exists x (\varphi \wedge \psi)$

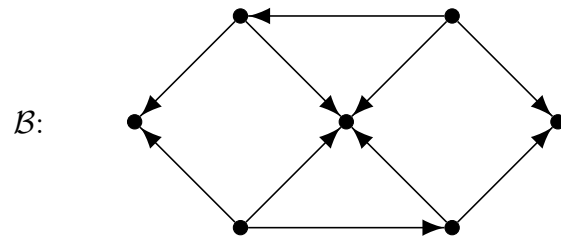
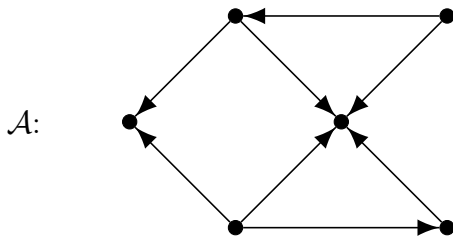
(ii) $(\forall x \varphi \vee \forall x \psi) \equiv \forall x (\varphi \vee \psi)$

(iv) $(\exists x \varphi \vee \exists x \psi) \equiv \exists x (\varphi \vee \psi)$

- (c) Beweisen Sie, dass ihre Antworten zu (ii) und (iv) in Aufgabenteil (b) korrekt sind.

Aufgabe 3:**(25 Punkte)**

Sei $\sigma := \{E/2\}$. Betrachten Sie die folgenden Graphen \mathcal{A} und \mathcal{B} :



- (a) Welches ist das kleinste m , so dass Spoiler eine Gewinnstrategie im m -Runden Ehrenfeucht-Fraïssé Spiel auf \mathcal{A} und \mathcal{B} hat? Begründen Sie Ihre Antwort, indem Sie eine Gewinnstrategie für Spoiler im m -Runden Ehrenfeucht-Fraïssé Spiel und eine Gewinnstrategie für Duplicator im $(m-1)$ -Runden Ehrenfeucht-Fraïssé Spiel beschreiben.
- (b) Finden Sie für Ihre Antwort m aus Teil (a) einen FO[σ]-Satz ψ der Quantorentiefe m , so dass $\mathcal{A} \models \psi$ und $\mathcal{B} \models \neg\psi$.

Aufgabe 4:

(25 Punkte)

Lesen Sie Kapitel 10 aus dem Buch „Learn Prolog Now!“.

Erstellen Sie eine Datei `blatt08.pl`, die mit der Zeile `:- ensure_loaded([a1]).` beginnt. Laden Sie in das selbe Verzeichnis die Datei `a1.pl` von der URL <http://www2.informatik.hu-berlin.de/logik/lehre/WS15-16/Logik/downloads/a1.pl>.

Achtung: Geben Sie die von Ihnen erstellte Datei `blatt08.pl` mit Ihren Lösungen der Teilaufgaben (b), (c) und (d) über das GOYA-System ab! **Außerdem gilt:** Lösungsansätze, die von SWI-Prolog nicht geladen werden können, werden nicht bewertet!

(a) Gegeben sei das folgende Prolog-Programm:

```
1 f(X) :- g(X).           5 h(1).
2 f(X) :- h(X), i(X).    6 h(2).
3 g(X) :- h(X), !, i(X). 7 i(1).
4 g(3).                  8 i(2).
```

Zeichnen Sie einen Suchbaum für die Anfrage `? f(X) . !`

(b) Schreiben Sie in `blatt08.pl` ein Prädikat `not_member/2`, so dass `not_member(X, L)` für einen Term `X` und eine Liste `L` genau dann erfüllt ist, wenn `X` mit *keinem* Element von `L` unifiziert werden kann. Verwenden Sie dabei abgesehen vom Cut und dem in SWI-Prolog vordefinierten Prädikat `fail/0` keine weiteren Prädikate, und insbesondere nicht `\=/2`.

Hinweis: Das Prädikat `fail/0` schlägt immer fehl. D.h., das Ziel `fail` in einer Anfrage löst immer das Backtracking aus.

(c) Führen Sie in `blatt08.pl` einen neuen Operator `<=>` für die Biimplikation \leftrightarrow ein, der den gleichen Typ und die gleiche Präzedenz wie der in `a1.pl` definierte Operator `=>` hat.

(d) Implementieren Sie in `blatt08.pl`, analog zu Beispiel 2.54 im Vorlesungsskript, Schritt 1 des Tseitin-Verfahrens. D.h., schreiben Sie ein Prädikat `tseitin/2`, so dass die Anfrage `tseitin(F, L)` für eine aussagenlogische Formel `F` eine Liste `L` aussagenlogischer Formeln ausgibt, die die folgenden Eigenschaften hat:

- Die Konjunktion der Formeln in der Liste `L` ist erfüllbarkeitsäquivalent zu `F`.
- Die Liste `L` enthält für jede Teilformel von `F` (abgesehen von Literalen) genau eine Formel.
- In jeder Formel aus `L` kommen höchstens 3 verschiedene Aussagensymbole vor.

Beispielsweise sollte die Anfrage `?- tseitin((p => ~q) \ / (~ (p /\ q) /\ r), L).` zu der Ausgabe

```
L = [x1, x1<=>x2\ / x3, x2<=> (p=> ~q), x3<=>x4/\r, x4<=> ~x5, x5<=>p/\q].
```

führen. Hierbei ist die konkrete Wahl der neuen Aussagensymbole sowie die Reihenfolge der Formeln in der Repräsentation der Menge unwesentlich.

Hinweise:

- Benutzen Sie zur Erzeugung neuer Aussagensymbole das in SWI-Prolog eingebaute Prädikat `gensym/2`. Das Prädikat `gensym/2` instantiiert bei dem Aufruf `gensym(x, A)` die Variable `A` mit einem Atom der Form `xn`, wobei eine Zahl `n` so gewählt wird, dass das Atom `xn` in diesem Lauf von SWI-Prolog noch nicht verwendet wurde.
- Benutzen Sie den in Teilaufgabe (c) definierten Operator `<=>`.
- Nutzen Sie ggf. Cut oder Negation. Führen Sie bei Bedarf Hilfsprädikate ein.