Formale Grundlagen der Informatik II
6. Übungsblatt

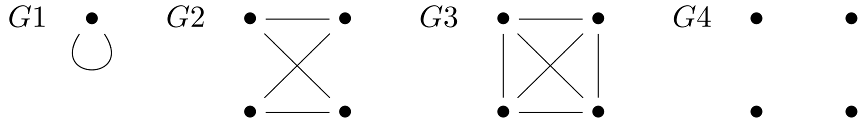
TECHNISCHE UNIVERSITÄT DARMSTADT

Fachbereich Mathematik
Prof. Dr. Martin Ziegler
Alexander Kreuzer
Carsten Rösnick

SS 2011
06.07.11

Minitest Lösung

Gegeben seien die folgenden ungerichteten Graphen $G = (V, E)$:



In welchem der obigen Graphen gilt welcher der nachfolgenden FO-Sätze?

(a) $G\boxed{3}$ $\quad \forall x \forall y \, (\neg(x = y) \leftrightarrow Exy)$

(b) $G\boxed{2}$ $\quad \exists x \exists y \exists z \, (\neg(x = y) \wedge \neg(y = z) \wedge Exy \wedge Eyz \wedge \neg Ezx)$

(c) $G\boxed{4}$ $\quad \exists x \exists y \neg \, (x = y) \wedge \forall x \forall y \, (\neg(x = y) \rightarrow \neg Exy)$

(d) $G\boxed{1}$ $\quad \exists x \forall y \, (x = y)$

Begründung: Die angegebenen FO-Sätze haben folgende Bedeutung:

(a) Je zwei verschiedene Knoten sind miteinander verbunden.

(b) Es gibt drei Knoten, die keinen Kreis bilden.

(c) Der Graph enthält keine Kante, aber mindestens zwei Knoten.

(d) Der Graph besteht aus nur einem Knoten.

Gruppenübung

Aufgabe G1

(a) Bestimmen Sie die Skolem-Normalform der folgenden Formel:

$$\varphi = \exists a \forall b \forall c \exists x \exists y \forall z \, (a \cdot b + x + z > c + y) \, .$$

(b) Geben Sie eine Interpretation der Skolem-Normalform von φ über der Struktur $\mathcal{N} = (\mathbb{N}, <^{\mathbb{N}}, +^{\mathbb{N}}, \cdot^{\mathbb{N}})$ der natürlichen Zahlen an.

(c) Geben Sie eine FO(S)-Formel $\varphi'(x)$ über der Signatur $S = (+, \cdot)$ an, die über der Struktur $\mathcal{N}$ äquivalent ist zu der Aussage: x ist eine Primzahl.

Aufgabe G2

Ein Pfad in einem Graph $\mathcal{G} = (V, E)$ ist eine Sequenz $\langle x_0, x_1, \ldots, x_n \rangle$ von Knoten, so dass

$$x_i E x_{i+1}$$

für alle $i < n$. Der Graph heißt *zusammenhängend*, wenn es für alle Paaren von Knoten (x, y) einen Pfad $\langle x_0, x_1, \ldots, x_n \rangle$ gibt, mit $x = x_0$ und $y = x_n$.
Zeigen Sie, dass es keine Formelmenge Γ in der Sprache der Graphen gibt, so dass $\mathcal{G} \models \Gamma$ genau dann wenn $\mathcal{G}$ zusammenhängend ist.

Aufgabe G3

Zeigen Sie, dass eine Theorie T, die beliebig grosse endliche Modelle hat, auch ein unendliches Modell besitzt.
Geben Sie auch einen Satz an, der endliche und unendliche Modelle hat, aber keine beliebig grossen endliche Modelle.

Hausübung

Aufgabe H1 (6 Punkte)

Welche der folgenden Eigenschaften der intendierten (d.h. hier: der auf $\mathbb{N}$ natürlichen) linearen Ordnung $(\mathbb{N}, <)$ lassen sich in $\mathrm{FO}(<)$ formalisieren, welche nicht?
Hier sind klare Begründungen (z.B. Formalisierungen oder Kompaktheitsargumente) verlangt.

(a) Jedes Element besitzt einen direkten Nachfolger.

(b) Es gibt kein letztes Element.

(c) Jedes Element hat nur endlich viele Vorgänger.

Aufgabe H2

Betrachten Sie einen endlichen, gerichteten Graphen $\mathcal{G} = (V, E)$.
In der Aufgabe G3 haben wir gesehen, dass es keinen FO-Satz $\varphi(x, y)$, der genau dann wahr ist, wenn es einen Pfad von dem Knoten x zu dem Knoten y gibt. (Gäbe es eine solche FO-Formel $\varphi(x, y)$, so würde $\Gamma = \forall x, y\, \varphi(x, y)$ ausdrücken, dass $\mathcal{G}$ zusammenhängend ist.)
Wir wollen in dieser Aufgabe untersuchen, wie man zu einem allgemeinen $\mathcal{G}$ eine zwei-stellige Relation Cxy (für **c**onnected), die das aussagt, hinzufügen kann, und was das für Programme bedeutet.

(a) Was bedeutet es für ein Prolog-Programm, bzw. für die Definition von Cxy in Prolog, dass es kein $\varphi(x, y)$ mit $Cxy \leftrightarrow \varphi(x, y)$ gibt?
Was bedeutet das für eine imperative Programmiersprache wie C oder Java?

(b) Betrachten Sie die Sätze

$$\begin{aligned} &\forall x \forall y\ (Exy \rightarrow Cxy) \\ &\forall x \forall y \forall z\ ((Exz \wedge Czy \rightarrow Cxy)\ . \end{aligned} \tag{1}$$

Was bedeuten diese Sätze umgangssprachlich?
Geben Sie einige Erweiterungen (d.h. Interpretationen der Relation C) des folgenden Graphen um eine Relation C an, die diese Sätze erfüllt.

$$\begin{tikzcd} & 1 \arrow[d] & 4 \arrow[d] \\ \mathcal{G}: & 2 \arrow[d] & 5 \\ & 3 & \end{tikzcd}$$

Welche Erweiterung ist die intendierte (=gewollte, natürliche) Erweiterung, also die Erweiterung, in der Cxy bedeutet, dass es einen Pfad von x nach y gibt?

(c) Sei nun $\mathcal{G} = (V, E) = (V, E^{\mathcal{G}})$ wieder ein beliebiger endlicher, gerichteter Graph.

Zeigen Sie, dass zwei Erweiterungen (V, E, C_1), (V, E, C_2), die die Sätze aus (1) erfüllen, auch $(V, E, C_1 \cap C_2)$ die Sätze aus (1) erfüllen.

Folgern Sie daraus, dass es eine minimale Erweiterung (V, E, C_{min}), die (1) erfüllt gibt, d.h. eine Erweiterung, so dass für eine jede andere solche Erweiterung (V, E, C_1) gilt $C_{min} \subseteq C_1$.

Wir versuchen nun diese C_{min} von unten zu approximieren.

(d) Betrachten Sie dafür die Operation

$$\mathsf{F}(X) := \{ (x, y) \mid (x, y) \in E^{\mathcal{G}} \text{ oder es gibt } z \in V \text{ mit } (x, z) \in E^{\mathcal{G}} \text{ und } (z, y) \in X \}.$$

Zeigen Sie, dass Φ monoton ist, d.h. das für alle $X, Y \subseteq V \times V$ gilt

$$X \subseteq Y \Longrightarrow \Phi(X) \subseteq \Phi(Y).$$

(e) Sei $C_0 := \emptyset$, $C_{n+1} = \mathsf{F}(C_n)$ und $C^* := \bigcup_{n=0}^{\infty} C_n$. Zeige, dass $C^* = C_{min}$ und damit, dass wir auf diese Weise die kleinste Erweiterung iterativ konstruiert haben.

Allgemein gilt, dass man für alle Sätze Φ der gleichen Form wie in (1) und zu jeder Struktur $\mathcal{A}$ eine minimale Erweiterung (um eine neue Relation R) finden kann, die dann auch von unten approximiert werden kann.[1]

(f) Implementieren Sie die Relation C mit Hilfe der Sätze (1) in Prolog. Nehmen Sie dabei an, dass der Graph durch ein Relation V für die Knoten und eine Relation E gegeben ist.

Z.B. wäre der Graphen aus der (b) gegen durch

```
v(1). v(2). v(3). v(4). v(5).
e(1,2). e(2,3). e(4,5).
```

Sie dürfen annehmen, dass der Graph azyklisch ist, d.h. es gibt keine Pfade von einen Knoten x zurück zu x.

Da Prolog Tiefensuche verwendet um nach erfüllbaren Prädikaten zu suchen, müssen sonst besondere Vorkehrungen getroffen werden, damit ein Zyklus nur einmal durchlaufen wird.

Datalog, ein Dialekt von Prolog, approximiert die Prädikate, wie wir es in dem Aufgabenteil (e) getan haben, deswegen kann dort C für alle Graphen wie in (1) definiert werden. Damit das funktioniert ist die Verwendung von Rekursion in Datalog eingeschränkt, im Westentlichen auf Sätze vom Typ wie in der Fußnote beschrieben.

[1] Genauer: Ein gleichungsfreier *nicht-negativer universeller Horn-Satz* ist ein Satz der Form

$$\forall x_1 \cdots \forall x_n [(\alpha_1 \wedge \cdots \wedge \alpha_m) \to \beta] ,$$

wobei $\alpha_1, \ldots, \alpha_m, \beta$ gleichungsfreie atomare Formeln sind. Wenn Φ aus Horn-Sätze, in denen $\alpha_1, \ldots, \alpha_n$ atomare Formeln über der Signatur von $\mathcal{A}$ und R sind und β aus R besteht.